

QCPU (Q模式) / QnACPU

编程手册

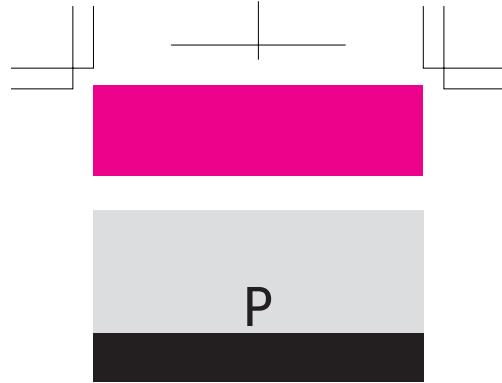
mitsubishi

(公共指令)



三菱可编程逻辑控制器

MELSEC-Q



QCPU(Q 模式)/QnACPU 编程手册(公共指令)



● 安全警告 ●

(在使用产品之前必须先阅读此警告)

在使用产品之前，请先阅读此手册和此手册中介绍的相关手册。请对安全问题给予足够重视以便能正确操作此产品。

请将此手册保存到一个安全的地方，以便在需要时可以方便地得到。请送给最终用户本手册的副本。

修订本

* 此手册编号在封底的左下角给出。

出版日期	* 手册编号	版本
2003 年 9 月	SH (NA) -080450CHN	初版

英文手册版本 SH-080039-G

此手册无任何工业产权，无如何其它类型产权，也无任何专利。三菱电机公司不对由于使用本手册中的说明而造成的任何可能包含的工业产权问题负责。

© * 1999 三菱电机株式会社

前言

感谢使用三菱电机 MELSEC-Q 系列 (Q 模式) 和 MELSEC-QnA 系列可编程逻辑控制器。

在使用这些产品之前, 请仔细阅读本手册, 使你能对所使用的 Q 系列 (Q 模式) /QnA 系列可编程逻辑控制器的功能和性能足够熟悉, 以便能保证正确使用该类产品。

应该送给最终用户此手册的一个副本。

目录

1. 综述	1 - 1 到 1 - 4
1.1 相关编程手册	1 - 1
1.2 缩写和通用名称	1 - 4
2. 指令表	2 - 1 到 2 - 50
2.1 指令分类	2 - 1
2.2 如何阅读指令表	2 - 2
2.3 顺序指令	2 - 4
2.3.1 触点指令	2 - 4
2.3.2 连接指令	2 - 5
2.3.3 输出指令	2 - 6
2.3.4 移位指令	2 - 6
2.3.5 主控制指令	2 - 7
2.3.6 终止指令	2 - 7
2.3.7 其它指令	2 - 7
2.4 基本指令	2 - 8
2.4.1 比较操作指令	2 - 8
2.4.2 算术操作指令	2 - 14
2.4.3 数据转换指令	2 - 19
2.4.4 数据移动指令	2 - 21
2.4.5 程序分支指令	2 - 23
2.4.6 程序执行控制指令	2 - 23
2.4.7 I/O 刷新指令	2 - 23
2.4.8 其它易于使用的指令	2 - 24
2.5 应用指令	2 - 25
2.5.1 逻辑操作指令	2 - 25
2.5.2 循环指令	2 - 28
2.5.3 移位指令	2 - 29
2.5.4 位处理指令	2 - 29
2.5.5 数据处理指令	2 - 30
2.5.6 结构体创建指令	2 - 33
2.5.7 数据表操作指令	2 - 34
2.5.8 缓冲区访问指令	2 - 35
2.5.9 显示指令	2 - 35
2.5.10 调试和错误诊断指令	2 - 36
2.5.11 字符串处理指令	2 - 37

2.5.12 特殊功能指令	2 - 40
2.5.13 数据控制指令	2 - 42
2.5.14 转换指令	2 - 43
2.5.15 时钟指令	2 - 44
2.5.16 外围设备指令	2 - 45
2.5.17 程序控制指令	2 - 45
2.5.18 其它指令	2 - 46
2.5.19 用于数据链接的指令	2 - 47
2.5.20 QCPU 指令	2 - 49
2.5.21 冗余系统指令(用于 Q4ARCPU)	2 - 50

3. 指令组态	3 - 1 到 3 - 29
---------	----------------

3.1 指令组态	3 - 1
3.2 指定数据	3 - 2
3.2.1 使用位数据	3 - 2
3.2.2 使用字(16 位)数据	3 - 3
3.2.3 使用双字数据(32 位)	3 - 5
3.2.4 使用实型数据	3 - 8
3.2.5 使用字符串数据	3 - 10
3.3 索引修改	3 - 11
3.4 间接指定	3 - 15
3.5 子集处理	3 - 17
3.6 编程警告(操作错误)	3 - 18
3.8 计数步号码	3 - 21
3.7 指令执行情况	3 - 22
3.9 当 OUT, SET/RST, 或者 PLS/PLF 指令使用同一设备时的操作	3 - 23
3.10 使用文件寄存器的警告	3 - 27
3.11 使用闪存的警告	3 - 29

4. 如何阅读指令	4 - 1 到 4 - 3
-----------	---------------

5. 顺序指令	5 - 1 到 5 - 55
---------	----------------

5.1 触点指令	5 - 2
5.1.1 操作开始, 串联, 并联(LD, LDI, AND, ANI, OR, ORI)	5 - 2
5.1.2 脉冲操作开始, 脉冲串联, 脉冲并联(LDP, LDF, ANDP, ANDF, ORP, ORF)	5 - 5
5.2 连接指令	5 - 7
5.2.1 梯形图串联和并联(ANB, ORB)	5 - 7
5.2.2 操作结果的进栈, 读, 出栈 (MPS, MRD, MPP)	5 - 9
5.2.3 操作结果取反 (INV)	5 - 13
5.2.4 操作结果脉冲转换 (MEP, MEF)	5 - 14
5.2.5 边沿继电器操作结果的脉冲转换(EGP, EGF)	5 - 16
5.3 输出指令	5 - 18
5.3.1 输出指令(包括定时器, 计数器和报警器) (OUT)	5 - 18
5.3.2 定时器(OUT T, OUTH T)	5 - 20
5.3.3 计数器(OUT C)	5 - 24

5.3.4 报警器输出 (OUT F)	5 - 26
5.3.5 置位设备(报警器除外) (SET)	5 - 28
5.3.6 复位设备(报警器除外) (RST)	5 - 30
5.3.7 置位和复位报警器 (SET F, RST F)	5 - 32
5.3.8 前沿和后沿输出 (PLS, PLF)	5 - 34
5.3.9 位设备输出倒置 (FF)	5 - 36
5.3.10 直接输出的脉冲转换 (DELTA, DELTAP)	5 - 38
5.4 移位指令	5 - 40
5.4.1 位设备移位 (SFT, SFTP)	5 - 40
5.5 主控制指令	5 - 42
5.5.1 置位和复位电机控制器 (MC, MCR)	5 - 42
5.6 结束指令	5 - 46
5.6.1 结束主循环程序 (FEND)	5 - 46
5.6.2 结束顺序程序 (END)	5 - 48
5.7 其它指令	5 - 50
5.7.1 顺序程序停止 (STOP)	5 - 50
5.7.2 空操作 (NOP, NOPLF, PAGE n)	5 - 52

6. 基本指令

6 - 1 到 6 - 133

6.1 比较操作指令	6 - 2
6.1.1 BIN 16 位数据比较 (=, <, >, <=, <, >=)	6 - 2
6.1.2 BIN 32 位数据比较 (D=, D<, D>, D<=, D<, D>=)	6 - 4
6.1.3 浮点十进制数据比较 (E=, E<, E>, E<=, E<, E>=)	6 - 6
6.1.4 字符串数据比较 (\$=, \$<, \$>, \$<=, \$<, \$>=)	6 - 8
6.1.5 BIN 块数据比较 (BKCOMP, BKCOMP □ P)	6 - 12
6.2 算术操作指令	6 - 16
6.2.1 BIN 16 位加和减操作 (+, +P, -, -P)	6 - 16
6.2.2 BIN 32 位加和减操作 (D+, D+P, D-, D-P)	6 - 20
6.2.3 BIN 16 位乘和除操作 (*, *P, /, /P)	6 - 24
6.2.4 BIN 32 位乘和除操作 (D*, D*P, D/, D/P)	6 - 26
6.2.5 BCD 4 位加和减操作 (B+, B+P, B-, B-P)	6 - 28
6.2.6 BCD 8 位加和减操作 (DB+, DB+P, DB-, DB-P)	6 - 32
6.2.7 BCD 4 位乘和除操作 (B*, B*P, B/, B/P)	6 - 36
6.2.8 BCD 8 位乘和除操作 (DB*, DB*P, DB/, DB/P)	6 - 38
6.2.9 浮点十进制数据的加和减 (E+, E+P, E-, E-P)	6 - 40
6.2.10 浮点十进制数据的乘和除 (E*, E*P, E/, E/P)	6 - 44
6.2.11 块的加和减 (BK+, BK+P, BK-, BK-P)	6 - 46
6.2.12 链接字符串 (\$+, \$+P)	6 - 49
6.2.13 增大和减小 16 位 BIN 数据 (INC, INCP, DEC, DECP)	6 - 53
6.2.14 增大和减小 32 位 BIN 数据 (DINC, DINCP, DDEC, DDECP)	6 - 55
6.3 数据转换指令	6 - 57
6.3.1 BIN 数据转换成 4 位和 8 位 BCD 码 (BCD, BCDP, DBCD, DBCDP)	6 - 57
6.3.2 BCD 4 位和 8 位数据转换成 BIN 数据 (BIN, BINP, DBIN, DBINP)	6 - 59
6.3.3 BIN 16 位和 32 位数据转换成浮点十进制数 (FLT, FLTP, DFLT, DFLTP)	6 - 61
6.3.4 浮点十进制数转换成 BIN 16 位和 32 位数据 (INT, INTP, DINT, DINTP)	6 - 63
6.3.5 BIN 16 位转换成 BIN 32 位数据 (DBL, DBLP)	6 - 65

6.3.6	BIN 32 位转换成 BIN 16 位数据 (WORD, WORDP)	6 - 66
6.3.7	BIN 16 和 32 位数据转换成格雷码 (GRY, GRYP, DGRY, DGRYP)	6 - 67
6.3.8	格雷码转换成 BIN 16 和 32 位数据 (GBIN, GBINP, DGBIN, DGBINP)	6 - 69
6.3.9	BIN 16 和 32 位数据的补充 (符号取反) (NEG, NEGP, DNEG, DNEGP)	6 - 71
6.3.10	浮点十进制数据的符号取反 (ENEG, ENEGP)	6 - 73
6.3.11	块 BIN 16 位数据转换成 BCD 4 位数据 (BKBCD, BKBCDP)	6 - 74
6.3.12	块 BCD 4 位数据转换成块 BIN 16 位数据 (BKBIN, BKBINP)	6 - 76
6.4	数据移动指令	6 - 78
6.4.1	16 位和 32 位数据移动 (MOV, MOVP, DMOV, DMOVP)	6 - 78
6.4.1	16 位和 32 位数据移动 (MOV, MOVP, DMOV, DMOVP)	6 - 80
6.4.3	字符串移动 (\$MOV, \$MOVP)	6 - 82
6.4.4	16 位和 32 位负数移动 (CML, CMLP, DCML, DCMLP)	6 - 84
6.4.5	16 位数据块移动 (BMOV, BMOVP)	6 - 87
6.4.6	相同的 16 位数据块移动 (FMOV, FMOVP)	6 - 89
6.4.7	16 位和 32 位数据交换 (XCH, XCHP, DXCH, DXCHP)	6 - 91
6.4.8	16 位数据块交换 (BXCH, BXCHP)	6 - 93
6.4.9	最高和最低字节交换 (SWAP, SWAPP)	6 - 95
6.5	程序分支指令	6 - 96
6.5.1	指针分支指令 (CJ, SCJ, JMP)	6 - 96
6.5.2	跳转到 END (GOEND)	6 - 99
6.6	程序执行控制指令	6 - 100
6.6.1	中断禁止/允许指令, 中断程序掩码 (DI, EI IMASK)	6 - 100
6.6.2	从中断程序中恢复 (IRET)	6 - 109
6.7	I/O 刷新指令	6 - 111
6.7.1	I/O 刷新 (RFS, RFSP)	6 - 111
6.8	其它使用方便的指令	6 - 113
6.8.1	计数器 1 相输入的增大或减小 (UDCNT1)	6 - 113
6.8.2	计数器 2 相输入的增大或减小 (UDCNT2)	6 - 115
6.8.3	教学定时器 (TTMR)	6 - 117
6.8.4	特殊功能定时器 (STMR)	6 - 119
6.8.5	循环表近路径循环控制 (ROTC)	6 - 122
6.8.6	斜坡信号 (RAMP)	6 - 124
6.8.7	脉冲密度测量 (SPD)	6 - 126
6.8.8	固定周期脉冲输出 (PLSY)	6 - 128
6.8.9	脉宽调制 (PWM)	6 - 130
6.8.10	矩阵输入 (MTR)	6 - 132

7. 应用指令

7 - 1 到 7 - 342

7.1	逻辑操作指令	7 - 2
7.1.1	16 位和 32 位数据的逻辑乘积 (WAND, WANDP, DAND, DANDP)	7 - 3
7.1.2	块逻辑乘积 (BKAND, BKANDP)	7 - 8
7.1.3	16 位和 32 位数据的逻辑和 (WOR, WORP, DOR, DORP)	7 - 10
7.1.4	块逻辑和操作 (BKOR, BKORP)	7 - 14
7.1.5	16 位和 32 位专有的 OR 操作 (WXOR, WXORP, DXOR, DXORP)	7 - 16
7.1.6	块专有的 OR 操作 (BKXOR, BKXORP)	7 - 20
7.1.7	16 位和 32 位数据非专有逻辑和操作 (WXNR, WXNRP, DXNR, DXNRP)	7 - 22
7.1.8	块非专有的逻辑和操作 (BKXNR, BKXNRP)	7 - 28

7.2 循环指令	7 - 30
7.2.1 16 位数据的右循环移动 (ROR, RORP, RCR, RCRP)	7 - 30
7.2.2 16 位数据的左循环移动 (ROL, ROLP, RCL, RCLP)	7 - 32
7.2.3 32 位数据的右循环移动 (DROR, DRORP, DRCR, DRCRP)	7 - 34
7.2.4 32 位数据的左循环移动 (DROL, DROLP, DRCL, DRCLP)	7 - 36
7.3 移位指令	7 - 38
7.3.1 16 位数据左移或者右移 n 位 (SFR, SFRP, SFL, SFLP)	7 - 38
7.3.2 n 位数据左移或者右移 1 位 (BSFR, BSFRP, BSFL, BSFLP)	7 - 40
7.3.3 n 字数据左移或者右移 1 字 (DSFR, DSFRP, DSFL, DSFLP)	7 - 42
7.4 位处理指令	7 - 44
7.4.1 字软元件的置位和复位 (BSET, BSETP, BRST, BRSTP)	7 - 44
7.4.2 位测试 (TEST, TESTP, DTEST, DTESTP)	7 - 46
7.4.3 位软元件的批复位 (BKRST, BKRSTP)	7 - 48
7.5 数据处理指令	7 - 50
7.5.1 16 位和 32 位数据查找 (SER, SERP, DSER, DSERP)	7 - 50
7.5.2 16 位和 32 位数据检查 (SUM, SUMP, DSUM, DSUMP)	7 - 54
7.5.3 从 8 位到 256 位解码 (DECO, DECOP)	7 - 56
7.5.4 从 256 位到 8 位编码 (ENCO, ENCO)	7 - 58
7.5.5 7 段解码 (SEG, SEGP)	7 - 60
7.5.6 16 位数据的 4 位编组 (DIS, DISP)	7 - 62
7.5.7 16 位数据的 4 位数据链接 (UNI, UNIP)	7 - 64
7.5.8 随机数据的分解或链接 (NDIS, NDISP, NUNI, NUNIP)	7 - 66
7.5.9 数据以字节为单位分解和链接 (WTOB, WTOBP, BTOW, BTOWP)	7 - 71
7.5.10 16 位和 32 位数据的最大值查找 (MAX, MAXP, DMAX, DMAXP)	7 - 75
7.5.11 16 位和 32 位数据的最小值查找 (MIN, MINP, DMIN, DMINP)	7 - 77
7.5.12 BIN 16 位和 32 位数据排序操作 (SORT, DSORT)	7 - 80
7.5.13 16 位数据的总数运算 (WSUM, WSUMP)	7 - 83
7.5.14 32 位数据的总数运算 (DWSUM, DWSUMP)	7 - 85
7.6 结构化程序指令	7 - 87
7.6.1 FOR 到 NEXT 指令循环 (FOR, NEXT)	7 - 87
7.6.2 FOR 到 NEXT 指令循环的强制结束 (BREAK, BREAKP)	7 - 89
7.6.3 子程序调用 (CALL, CALLP)	7 - 91
7.6.4 从子程序中返回 (RET)	7 - 95
7.6.5 子程序输出 OFF 调用 (FCALL, FCALLP)	7 - 96
7.6.6 程序文件中调用子程序 (ECALL, ECALLP)	7 - 100
7.6.7 程序文件中子程序输出 OFF 调用 (EFCALL, EFCALLP)	7 - 104
7.6.8 子程序调用 (XCALL)	7 - 108
7.6.9 刷新指令 (COM)	7 - 112
7.6.10 全部梯形图的索引修改 (IX, IXEND)	7 - 118
7.6.11 在全部梯形图的索引修改中修改值的指定 (IXDEV, IXSET)	7 - 126
7.7 数据表操作指令	7 - 131
7.7.1 写数据到数据表 (FIFW, FIFWP)	7 - 131
7.7.2 从表中读取最旧的数据 (FIFR, FIFRP)	7 - 133
7.7.3 从数据表中读取最新的数据 (FPOP, FPOPP)	7 - 135
7.7.4 从数据表中删除和在数据表中插入数据 (FDEL, FDELP, FINS, FINS)	7 - 137
7.8 缓冲区访问指令	7 - 140
7.8.1 从智能功能模块/特殊功能模块中读取 1/2 个字数据 (FROM, FROMP, DFRO, DFROP)	7 - 140

7.8.2 写 1/2 个字数据到智能功能模块/特殊功能模块 (TO, TOP, DTO, DTOP)	7 - 143
7.9 显示指令	7 - 146
7.9.1 打印 ASCII 编码指令 (PR)	7 - 146
7.9.2 打印注释指令 (PRC)	7 - 149
7.9.3 ASCII 编码 LED 显示指令 (LED)	7 - 154
7.9.4 用于注释的 LED 显示指令 (LEDC)	7 - 156
7.9.5 错误显示和报警器复位指令 (LEDR)	7 - 158
7.10 调试和故障诊断指令	7 - 161
7.10.1 特殊格式故障检查 (CHKST, CHK)	7 - 161
7.10.2 改变 CHK 指令的检查格式 (CHKCIR, CHKEND)	7 - 165
7.10.3 置位和复位状态闭锁 (SLT, SLTR)	7 - 173
7.10.4 置位和复位采样跟踪 (STRA, STRAR)	7 - 175
7.10.5 执行, 设置, 和复位程序跟踪 (PTRAE, PTRAXEP, PTRAR)	7 - 177
7.11 字符串处理指令	7 - 179
7.11.1 BIN 16 位或 32 位转换成十进制 ASCII (BINDA, BINDAP, DBINDA, DBINDAP)	7 - 179
7.11.2 BIN 16 位或 32 位数据转换成十六进制 ASCII (BINHA, BINHAP, DBINHA, DBINHAP)	7 - 182
7.11.3 BCD4 位和 8 位转换成十进制 ASCII 数据 (BCDDA, BCDDAP, DBCDDA, DBCDDAP)	7 - 185
7.11.4 十进制 ASCII 转换成 BIN 16 位和 32 位数据 (DABIN, DABINP, DDABIN, DDABINP)	7 - 188
7.11.5 十六进制 ASCII 转换成 BIN 16 位和 32 位数据 (HABIN, HABINP, DHABIN, DHABINP)	7 - 191
7.11.6 十进制 ASCII 转换成 BCD4 位或 8 位数据 (DABCD, DABCDP, DDABCD, DDABCDP)	7 - 193
7.11.7 读设备注释数据 (COMRD, COMRDP)	7 - 196
7.11.8 字符串长度探测 (LEN, LENP)	7 - 201
7.11.9 BIN 16 位或 32 位转换成字符串 (STR, STRP, DSTR, DSTRP)	7 - 203
7.11.10 字符串转换成 BIN 16 位或 32 位数据 (VAL, VALPP, DVAL, DVALP)	7 - 209
7.11.11 浮点十进制数据转换成字符串数据 (ESTR, ESTRP)	7 - 214
7.11.12 字符串转换成浮点十进制数据 (EVAL, EVALP)	7 - 221
7.11.13 十六进制 BIN 转换成 ASCII (ASC, ASCP)	7 - 225
7.11.14 ASCII 转换成十六进制 BIN (HEX, HEXP)	7 - 227
7.11.15 从左边或右边将字符串数据展开 (RIGHT, RIGHTP, LEFT, LEFTP)	7 - 229
7.11.16 在字符串中随机选取和替代 (MIDR, MIDRP, MIDW, MIDWP)	7 - 232
7.11.17 字符串搜索 (INSTR, INSTRP)	7 - 236
7.11.18 浮点十进制数转换成 BCD (EMOD, EMODP)	7 - 238
7.11.19 BCD 格式的数据转换成浮点十进制数据 (EREXP, EREXP)	7 - 240
7.12 特殊功能指令	7 - 242
7.12.1 浮点十进制数据的 SIN 操作 (SIN, SINP)	7 - 242
7.12.2 浮点十进制数据的 COS 操作 (COS, COSP)	7 - 244
7.12.3 浮点十进制数据的 TAN 操作 (TAN, TANP)	7 - 246
7.12.4 浮点十进制数据的 SIN^{-1} 操作 (ASIN, ASINP)	7 - 248

7.12.5	浮点十进制数据的 \cos^{-1} 操作 (ACOS, ACOSP)	7 - 250
7.12.6	浮点十进制数据的 \tan^{-1} 操作 (ATAN, ATANP)	7 - 252
7.12.7	将浮点十进制角度转换成弧度 (RAD, RADP)	7 - 254
7.12.8	将浮点十进制弧度转换成角度 (DEG, DEGP)	7 - 256
7.12.9	浮点十进制数据的方根操作 (SQR, SQRP)	7 - 259
7.12.10	浮点十进制数据的指数操作 (EXP, EXPP)	7 - 261
7.12.11	浮点十进制数据的自然对数操作 (LOG, LOGP)	7 - 263
7.12.12	随机自然数的产生和级数更新 (RND, RNDP, SRND, SRNDP)	7 - 265
7.12.13	BCD 4 位和 8 位方根 (BSQR, BSQRP, BDSQR, BDSQRP)	7 - 267
7.12.14	BCD 字节 SIN 操作 (BSIN, BSINP)	7 - 270
7.12.15	BCD 字节 COS 操作 (BCOS, BCOSP)	7 - 272
7.12.16	BCD 字节 TAN 操作 (BTAN, BTANP)	7 - 273
7.12.17	BCD 字节 \sin^{-1} 操作 (BASIN, BASINP)	7 - 275
7.12.18	BCD 字节 \cos^{-1} 操作 (BACOS, BACOSP)	7 - 277
7.12.19	BCD 字节 \tan^{-1} 操作 (BATAN, BATANP)	7 - 279
7.13	数据控制指令	7 - 281
7.13.1	BIN 16 位和 BIN 32 位数据的最高和最低限控制 (LIMIT, LIMITP, DLIMIT, DLIMITP)	7 - 281
7.13.2	BIN 16 位和 32 位死区带控制 (BAND, BANDP, DBAND, DBANDP)	7 - 284
7.13.3	BIN 16 位和 BIN 32 位数据的范围控制 (ZONE, ZONEP, DZONE, DZONEP)	7 - 287
7.14	文件寄存器转换指令	7 - 290
7.14.1	转换文件寄存器号码 (RSET, RSETP)	7 - 290
7.14.1	通过设定使用文件寄存器 (QDRSET, QDRSETP)	7 - 292
7.14.3	设置文件用于注释 (QCDSET, QCDSETP)	7 - 294
7.15	时钟指令	7 - 296
7.15.1	读取时钟数据	7 - 296
7.15.2	写时钟数据 (DATEWR, DATEWRP)	7 - 300
7.15.3	时钟数据加操作 (DATE+, DATE+P)	7 - 304
7.15.4	时钟数据减操作 (DATE-, DATE-P)	7 - 306
7.15.5	改变时间数据的格式 (SECOND, SECONDP, HOUR, HOURP)	7 - 308
7.16	外围设备指令	7 - 310
7.16.1	显示信息到外围设备 (MSG)	7 - 310
7.16.2	来自外围设备的键盘输入 (PKEY)	7 - 312
7.17	程序控制指令	7 - 314
7.17.1	程序等待指令 (PSTOP, PSTOPP)	7 - 315
7.17.2	程序输出 OFF 等待指令 (POFF, POFFP)	7 - 316
7.17.3	程序扫描执行登记指令 (PSCAN, PSCANP)	7 - 318
7.17.4	程序低速执行登记指令 (PLOW, PLOWP)	7 - 320
7.17.5	程序执行状态检查指令 (PCHK)	7 - 322
7.18	其它指令	7 - 324
7.18.1	复位看门狗定时器 (WDT, WDTP)	7 - 324
7.18.2	定时脉冲生成 (DUTY)	7 - 326
7.18.3	时间检查指令 (TIMCHK)	7 - 328
7.18.4	直接从文件寄存器中读取 1-字节 (ZRRDB, ZRRDBP)	7 - 329
7.18.5	文件寄存器中直接写入 1-字节 (ZRWRB, ZRWRBP)	7 - 331
7.18.6	间接地址读操作 (ADRSET, ADRSETP)	7 - 333
7.18.7	键盘数字键输入 (KEY)	7 - 334
7.18.8	索引寄存器的批量存盘或恢复 (ZPUSH, ZPUSHP, ZPOP, ZPOPP)	7 - 338
7.18.9	对 EEPROM 文件寄存器进行批量写操作 (EROMWR, EROMWRP)	7 - 342

8. 用于数据链接的指令	8 - 1 到 8 - 106
--------------	-----------------

8.1	网络刷新指令	8 - 7
8.1.1	网络刷新 (ZCOM)	8 - 7
8.2	QnA 链接的专用指令	8 - 15
8.2.1	从其它站中读取字设备数据 (READ)	8 - 15
8.2.2	从其它站中读取字设备数据 (SREAD)	8 - 21
8.2.3	将设备数据写到 MELSECNET/10 网络中的站 (WRITE)	8 - 27
8.2.4	写设备数据到其它站 (SWRITE)	8 - 34
8.2.5	发送数据到其它站 (SEND)	8 - 41
8.2.6	从其它站接收数据 (RECV)	8 - 49
8.2.7	来自其它站的暂时请求 (读/写时钟数据, 远程 RUN/STOP) (REQ)	8 - 55
8.2.8	从远程 I/O 站中的特殊功能模块中读取数据 (ZNFR)	8 - 67
8.2.9	写数据到远程 I/O 站中的特殊功能模块 (ZNT0)	8 - 72
8.3	用于 A-系列兼容链接的指令	8 - 77
8.3.1	从其它站读取设备数据 (MELSECNET/10) (ZNRD)	8 - 77
8.3.2	从本地站读取设备数据 (MELSECNET) (ZNRD)	8 - 81
8.3.3	写设备数据到其它站 (MELSECNET/10) (ZNWR)	8 - 84
8.3.4	写数据到本地站设备 (MELSECNET) (ZNWR)	8 - 88
8.3.5	从远程 I/O 站的特殊功能模块中读取数据 (MELSECNET) (RFRP)	8 - 91
8.3.6	写数据到远程 I/O 站中的特殊功能模块中 (MELSECNET) (RTOP)	8 - 95
8.4	路由信息读/写	8 - 99
8.4.1	读路由信息 (RTREAD)	8 - 99
8.4.2	登记路由信息 (RTWRITE)	8 - 103

9. QCPU 指令	9 - 1 到 9 - 53
------------	----------------

9.1	读模块信息 (UNIRD (P))	9 - 2
9.2	跟踪置位/复位 (TRACE, TRACER)	9 - 6
9.3	写数据到指定文件 (FWRITE)	9 - 8
9.4	从指定文件中读取数据 (FREAD)	9 - 16
9.5	从存储卡中载入程序 (PLOADP)	9 - 27
9.6	从程序存储器中上载程序 (PUNLOADP)	9 - 30
9.7	载入+上载 (PSW 附录)	9 - 33
9.8	文件寄存器的高速块传输 (RBMov (P))	9 - 36
9.9	写数据到主站 CPU 的共享存储器	9 - 39
9.9.1	写数据到主站 CPU 的共享存储器 (S.T0 (P))	9 - 41
9.9.2	写数据到主站 CPU 的共享存储器 (T0)	9 - 44
9.10	从另一个站的 CPU 共享存储器中读取数据	9 - 47
9.10.1	从另一个站的 CPU 共享存储器中读取数据 (FROM (P))	9 - 48
9.11	选择性刷新指令 (COM)	9 - 51

10. 冗余系统指令 (用于 Q4ARCPU)	10 - 1 到 10 - 14
-------------------------	------------------

10.1	在 CPU 启动时操作模式设定指令 (S. STMODE)	10 - 2
10.2	CPU 切换时间操作模式设定指令 (S. CGMODE)	10 - 4
10.3	数据跟踪指令 (S. TRUCK)	10 - 6
10.4	缓冲存储区批量刷新指令 (S. SPREF)	10 - 10

11. 错误代码	11- 1 到 11 - 54
----------	-----------------

11.1 如何读取错误代码	11 - 1
11.2 错误代码列表	11 - 2
11.2.1 基本型 QCPU 的错误代码列表	11 - 2
11.2.2 高性能型 QCPU/QnACPU 的错误代码列表	11 - 16
11.2.3 过程 CPU 的错误代码列表	11 - 36
11.3 复位一个错误	11 - 54

附录	附录 - 1 到 附录 - 172
----	-------------------

附录 1 操作处理时间	附录 - 1
1.1 定义	附录 - 1
1.2 基本型 QCPU 的操作处理时间	附录 - 2
1.3 高性能型 QCPU/过程 CPU/QnACPU 的操作处理时间	附录 - 18
附录 2 不同 CPU 的性能比较	附录 - 45
2.1 Q/QnACPU 和 AnNCPU, AnACPU, 以及 AnUCPU 的比较	附录 - 45
2.1.1 可用设备软元件软元件	附录 - 45
2.1.2 I/O 控制模式	附录 - 46
2.1.3 可被指令使用的数据	附录 - 47
2.1.4 定时器对比	附录 - 47
2.1.5 计数器对比	附录 - 48
2.1.6 显示指令对比	附录 - 48
2.1.7 指定格式发生变化的指令 (不包括 AnACPU 和 AnUCPU 的专用指令)	附录 - 49
2.1.8 AnACPU 和 AnUCPU 的专用指令	附录 - 50
2.1.9 可以在通用目标模式中编程的指令	附录 - 50
附录 3 特殊继电器列表	附录 - 51
3.1 基本型 QCPU 的特殊继电器列表	附录 - 51
3.2 高性能型 QCPU/QnACPU 的特殊继电器列表	附录 - 56
3.3 过程 CPU 的特殊继电器列表	附录 - 78
附录 4 特殊寄存器列表	附录 - 96
4.1 基本型 QCPU 的特殊寄存器列表	附录 - 96
4.2 高性能型 QCPU/QnACPU 的特殊寄存器列表	附录 - 108
4.3 过程 CPU 的特殊寄存器列表	附录 - 144
附录 5 应用程序实例	附录 - 172
5.1 执行操作 X^n 的程序, $\sqrt[n]{X}$ $\sqrt[n]{X}$	附录 - 172

手册

下表中列出了与 Q/QnACPU 相关的手册名单。请订购你需要的手册。

相关手册

手册名称	手册编号 (模块编码)
Q 系列精简模式 CPU 用户手册(硬件设计, 维护保养篇)。 描述了 CPU 模块的规范, 供电模块, 基本单元, 和扩展电缆。 (单独订购)	SH-080333C (13JR43)
Q 系列精简模式 CPU 用户手册(功能描述, 编程基础) 功能描述, 编程方法, 和基本模块 QCPU(Q 模式)的程序创建设备。 (单独订购)	SH-080331C (13JR44)
Q 系列 CPU 用户手册(硬件设计, 维护和保养篇) 描述了 CPU 模块的规范, 供电模块, 基本单元, 扩展电缆, 和存储卡。 (单独订购)	SH-080233 (13JL97)
Q 系列 CPU 用户手册(功能描述, 编程基础) 功能描述, 编程方法, 和高性能模型 QCPU(Q 模式)的程序创建设备。 (单独订购)	SH-080232 (13JL98)
Process CPU Users Manual (Hardware Design, Maintenance and Inspection) 描述了 CPU 模块的规范, 供电模块, 基本单元, 扩展电缆, 和存储卡。 (单独订购)	SH-080314E
Process CPU Users Manual (Functions Explanation, Programming Fundamentals) 功能描述, 编程方法, 创建程序所需设备。 (单独订购)	SH-080315E
QCPU/QnACPU 编程手册(SFC 控制指令篇) 描述了系统组态, 性能规范, 功能, 编程, 调试, 和 MELSAP3 的错误代码。 (单独订购)	SH-080283C
SW5D5C-GPPW 操作手册(MELSAP-L) 描述了系统组态, 性能规范, 功能, 编程, 调试, 和的错误代码。 (单独订购)	SH-080412C
Q/QnA 编程手册(PID 控制指令编) 描述了 PID 控制的专用指令。 (单独订购)	SH-080240C
QCPU(Q mode)/QnACPU Programming Manual(PID Control Instructions) 解释了在结构化文本语言中编程的方法。 (可选)	SH-080366E
QCPU(Q mode) Programming Manual(Structured Text) 描述了用于过程控制的专用指令。 (单独订购)	SH-080316E
QnACPU Guidebook 适用于第一次使用 QnACPU 的用户。描述了从创建程序到将创建好的程序写入 CPU 模块, 直到调试的一切步骤。 还说明了如何最有效地使用 QnACPU。	IB-66606
Q2A(S1)/Q3A/Q4ACPU User's Manual 描述了 Q2ACPU(S1), Q3ACPU, 和 Q4ACPU 的性能, 功能函数, 和处理操作。以及规格说明和供电模块、存储卡 和基本单元的处理操作。 (单独订购)	IB-66608
CPU 模式 Q2ASCPU 用户手册 描述了 Q2ASCPU, Q2ASCPU-S1, Q2ASHCPU, 和 Q2ASHCPU-S1 的性能, 功能函数, 和处理操作。以及规格说明、供 电模块和存储卡, 和基本单元的处理操作。 (单独订购)	SH-080215C

手册名称	手册编号 (模块编码)
<p>Q4ARCPU User's Manual</p> <p>描述了 Q4ARCPU 的性能, 功能, 和使用方法。还描述了总线开关模块, 系统管理模块, 供电模块, 存储卡, 和基本单元的规范和使用方法。</p> <p>(单独订购)</p>	IB-66685
<p>QnACPU Programming Manual (Fundamentals)</p> <p>描述了如何创建程序, 设备名称, 参数, 和程序类型。</p> <p>(单独订购)</p>	IB-66614
<p>QnACPU Programming Manual (Special Function Module)</p> <p>描述了当使用 Q2ACPU (S1), Q3ACPU, 和 Q4ACPU 时可用于特殊功能模块的专用指令。</p> <p>(单独订购)</p>	SH-4013
<p>QnACPU Programming Manual (AD57 Instructions)</p> <p>描述了当使用 Q2ACPU (S1), Q3ACPU, 或 Q4ACPU 时, 可用于控制 AD57 (S1) 型 CRT 控制器模块的专用指令集。</p> <p>(单独订购)</p>	IB-66617
<p>QnACPU Programming Manual (PID Control Instructions)</p> <p>描述了当使用 Q2ACPU (S1), Q3ACPU, 或 Q4ACPU 时, 可用于 PID 控制的专用指令集。</p> <p>(单独订购)</p>	IB-66618
<p>QnACPU Programming Manual (SFC)</p> <p>描述了 MELSAP3 的系统组态, 性能规范, 功能, 编程, 调试, 和错误代码。</p> <p>(单独订购)</p>	IB-66619
<p>For QnA/Q4AR MELSECNET/10 Network System Reference Manual</p> <p>描述了 MELSECNET/10 的一般概念, 规范, 组件名称和设置。</p> <p>(单独订购)</p>	IB-66690
<p>type MELSECNET, MELSECNET/B Data Link System Reference Manual</p> <p>描述了 MELSECNET (II) 和 MELSECNET/B 的一般概念, 规范, 和组件名称及设置。</p> <p>(单独订购)</p>	IB-66350
<p>GX Developer Ver. 7/simulator Ver. 6 操作手册。</p> <p>描述了 GX Developer Version 7 的在线功能函数, 包括编程过程, 打印程序, 监视程序, 和调试程序。</p> <p>(单独订购)</p>	SH-080311C
<p>Type SW2IVD-GPPQ software package OPERATING MANUAL (Offline)</p> <p>描述了当使用 SW2IVD-GPPQ 时如何创建程序和打印出数据, 以及 SW2IVD-GPPQ 的离线功能, 如文件维护等。</p> <p>(与产品一起提供)</p>	IB-66774
<p>Type SW2IVD-GPPQ software package OPERATING MNUAL (Online)</p> <p>描述了 SW2IVD-GPPQ 的在线功能, 包括监视和调试的方法。</p> <p>(与产品一起提供)</p>	IB-66775
<p>Type SW2IVD-GPPQ software package OPERATING MANUAL (SFC)</p> <p>描述了 SFC 功能函数, 如 SFC 程序的编辑和监视。</p> <p>(与产品一起提供)</p>	IB-66776

[illegible]

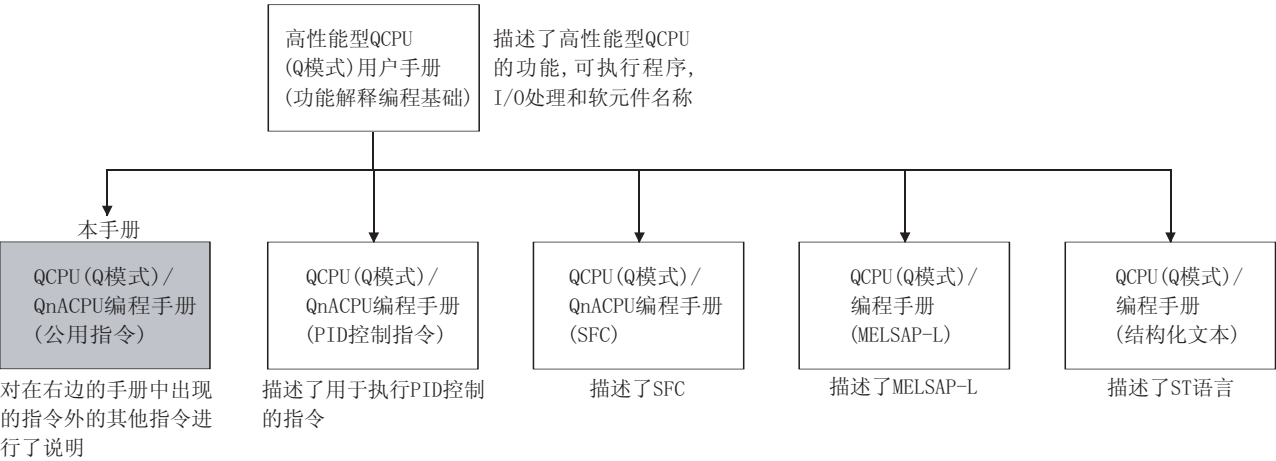
1. 一般概述

本手册描述 QCPU，QnACPU 和 Q2AS (H) CPU (S1) 的公用指令，这些指令用于在 QCPU (Q 模式)，QnACPU 和 Q2AS (H) CPU (S1) 上进行编程。公用指令是除了用于特殊功能模块的指令之外的其他所有指令，特殊功能模块指令包括 AJ71QC24 和 AJ71PT32-S3，用于 AD57 的指令，用于 PID 控制，和用于 MELSAP3 和 MELSAP-L 的指令。

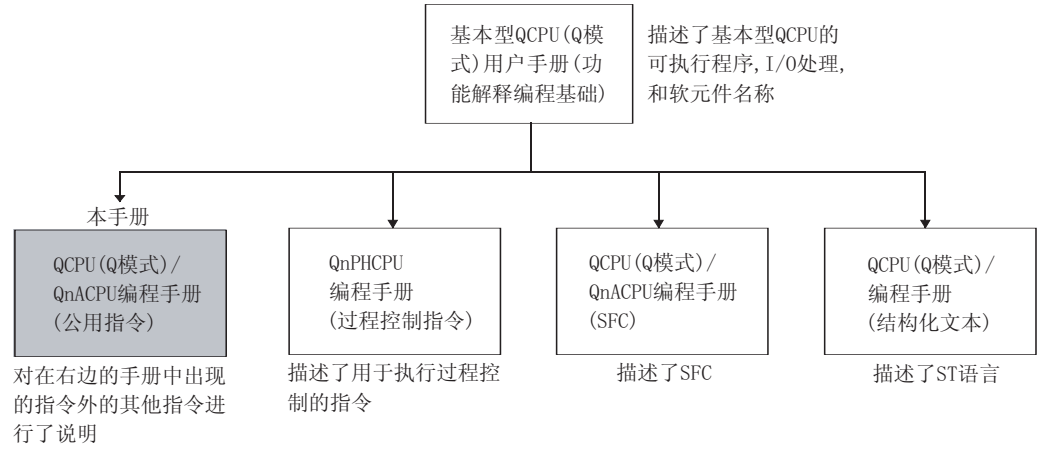
1.1 相关编程手册

在阅读本手册之前，请检查程序，I/O 处理，在 CPU 模块用户手册或 QnACPU 编程手册 (基础篇) 中和 CPU 模块一起使用的软元件。

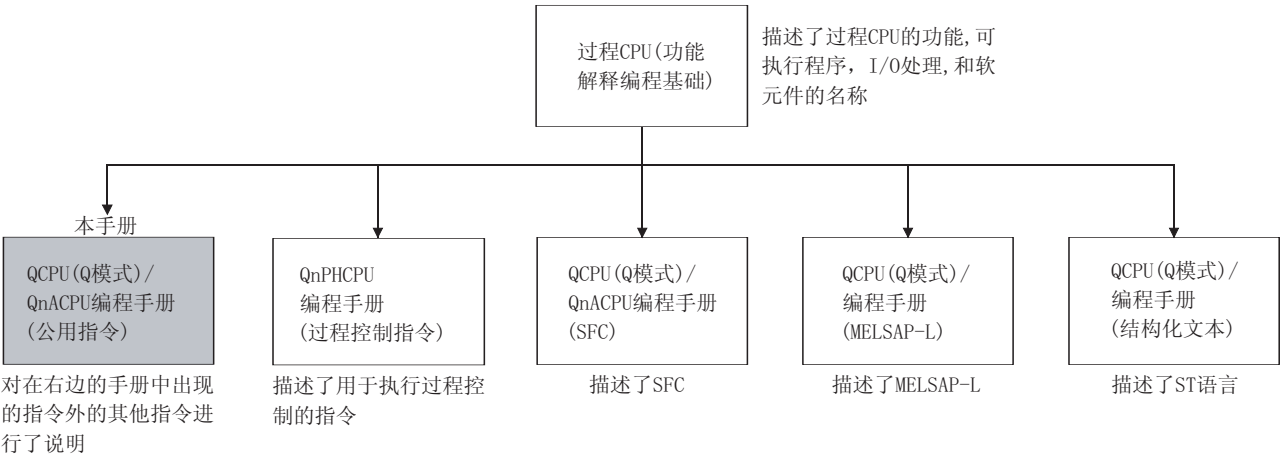
(1) Q02 (H) CP，UQ06HCPU，Q12HCPU，Q25HCPU



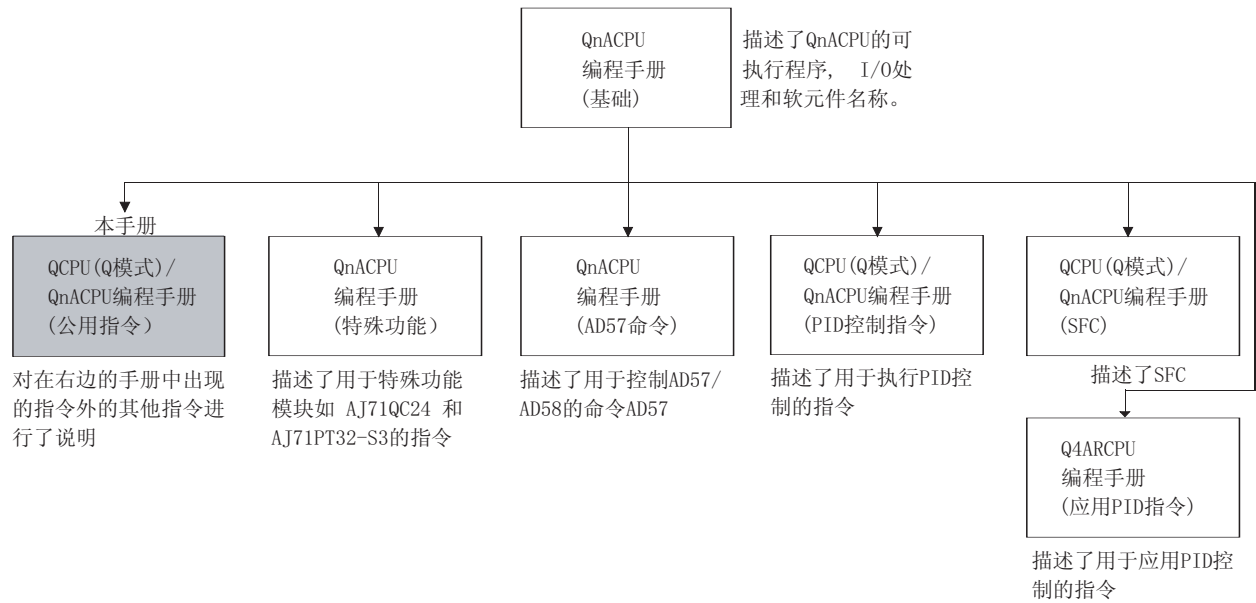
(2) Q00JCPU, Q00CPU, Q01CPU



(3) Q12PHCPU, Q25PHCPU



(4) Q2ACPU, Q3ACPU, Q4ACPU, Q4ARCPU, Q2AS (H) CPU



11.2 缩略语和通用名称模块名称的缩写

如下所示

模块类型名称	缩略语	在表中的缩略语	通用名称
Q00JCPU PLC CPU Q00CPU PLC CPU Q01CPU PLC CPU Q02CPU PLC CPU Q02HCPU PLC CPU Q06HCPU PLC CPU Q12HCPU PLC CPU Q25HCPU PLC CPU Q12PHCPU PLC CPU Q25PHCPU PLC CPU	QCPU	—	CPU
Q00JCPU PLC CPU Q00CPU PLC CPU Q01CPU PLC CPU	基本型 QCPU	基本型 QCPU	
Q02CPU PLC CPU Q02HCPU PLC CPU Q06HCPU PLC CPU Q12HCPU PLC CPU Q25HCPU PLC CPU	高性能型 QCPU	高性能型 QCPU	
Q12PHCPU PLC CPU Q25PHCPU PLC CPU	过程 CPU	过程 CPU	
Q2ACPU (S1) PLC CPU Q3ACPU PLC CPU Q4ACPU PLC CPU	QnACPU	QnA	
Q2ASCPU (S1) PLC CPU Q2ASHCPU (S1) PLC CPU	Q2ASCPU	Q2AS	
Q4ARCPU PLC CPU	Q4ARCPU	Q4AR	
MELSECNET/HNetwork 系统 (MELSECNET/H 模块)	MELSECNET/H	MELSECNET/10 (H)	—
MELSECNET/HNetwork 系统 (MELSECNET/H 模块)			
MELSECNET/10Network 系统	MELSECNET/10		
Ethernet 接口模块	以太网接口模块	以太网接口模块	—
控制和通信链接系统主/本地模块	CC-Link 模块	CC-Link 模块	—

2. 指令表

2.1 指令类型

CPU 模块指令的主要类型包括顺序指令，基本指令，应用指令，数据链接指令，QCPU 指令和冗余系统指令。这些指令类型列在下面的表 2.1。

表 2.1 指令类型

指令类型	含意	指令类型
顺序指令	触点指令	5
	连接指令	
	输出指令	
	移位指令	
	主站控制指令	
	终止指令	
	其它指令	
基本指令	比较操作指令	6
	算术操作指令	
	BCD BIN 转换指令	
	数据转移指令	
	程序分支指令	
	程序运行控制指令	
	I/O 刷新	
应用指令	其它使用方便的指令	7
	逻辑操作指令	
	循环指令	
	移位指令	
	位处理指令	
	数据处理指令	
	结构体创建指令	
	表操作指令	
	缓冲存储区访问指令	
	显示指令	
	调试和故障诊断指令	
	字符串处理指令	
	特殊功能指令	
	数据控制指令	
	交换指令	
	时钟指令	
	外围设备指令	
	程序指令	
	其它指令	
数据链接指令	链接刷新指令	8
	专门用于 QnA 链接的指令	
	用于 A-系列-兼容链接的指令	
	路由信息读/写指令	
QCPU 指令	用于 QCPU 的指令	9
冗余系统指令	用于 Q4ARCPU 的指令	10

2.2 如何阅读指令表

从章节 2.3 到章节 2.6 的指令表是按照下列形式制作的：

表 2.2 如何阅读指令表

种类	指令符号	符号	详细处理过程	执行条件	基本步数	子集	参见章节
BIN16 位加法和减法操作	+		● (D)+(S)→(D)		3	●	6-16
	+P						
	+		● (S1)+(S2)→(D)		4	●	6-18
	+P						

↑

①

↑

②

↑

③

↑

④

↑

⑤

↑

⑥

↑

⑦

↑

⑧

描述

- ①..... 按照指令的应用对它们进行分类。
- ②..... 表示增加到程序中的指令的指令符号。
指令代码是围绕 16 位指令建立的。下列符号用来标志只在由 OFF 变成 ON 的前沿被执行 32 位指令，实数指令，和字符串指令：
 - 32 位指令..... 在指令的第一行增加字母“D”
例子 + ➡ D+
 ↓ ↓
 16位指令 32位指令
 - 这些指令只在由 OFF 变成 ON 的前沿被执行
 在指令的最后增加字母“P”
例子 + ➡ +P
 ↓ ↓
 当为ON时执 在由OFF变为ON的
 行此指令 前沿时执行此指令
 - 实数指令 在指令的第一行增加字母“E”
例子 + ➡ E+
 ↓
 实数指令
 - 字符串指令 在指令的第一行增加一个美元符号“\$”
例子 + ➡ \$+
 ↓
 字符串指令

③..... 在梯形图上显示符号图表

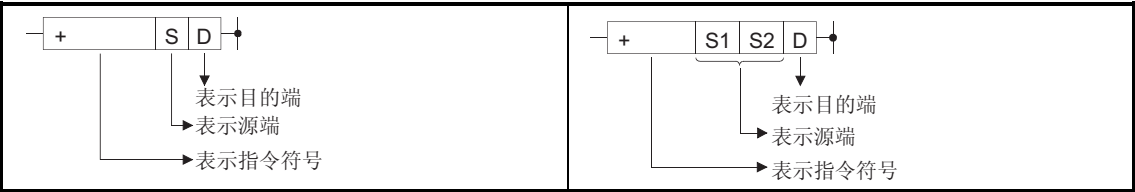


图 2.1 在梯形图上显示符号图表

目的端.....表示数据在操作后被送往何处
源端.....操作前存储数据的地方

④..... 表示由单个指令执行的操作的类型

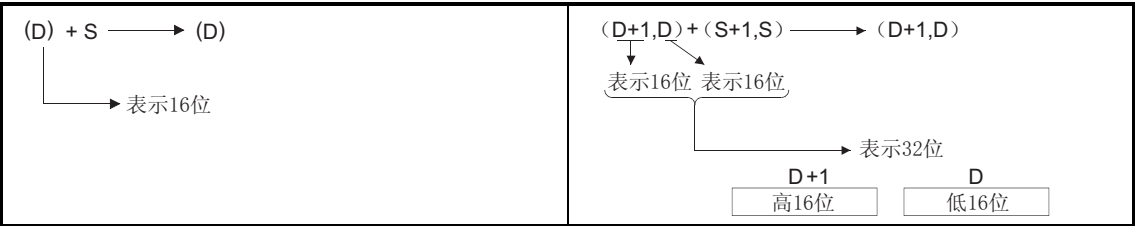


图 2.2 由单个指令执行的操作的类型

⑤..... 单个指令的执行情况细节如下所示：

符号	执行情况
无记录符号	指令在一般条件下执行，而不管指令前的 ON/OFF 状态。 如果前面的状态是 OFF，指令将执行 OFF 处理
	条件为 ON 时执行；只有当前面的状态为 ON 时才执行指令。 如果前面的状态为 OFF，指令将不执行，并且不执行任何处理
	一旦为 ON 时执行；只有前面的状态由 OFF 变为 ON 时在前沿处执行指令。此执行操作之后，即使条件保持为 ON，指令也不被执行并且不执行任何处理。
	当状态为 OFF 时执行；只有当前面的条件为 OFF 时指令才执行。 如果前面的状态为 ON，指令不被执行，并且不执行任何处理
	一旦为 OFF 时执行；当前面的条件由 ON 变成 OFF 时在后沿处执行指令。此执行操作之后，即使条件保持为 OFF 指令也不被执行，并且不会执行任何处理。

- ⑥..... 表示单条指令的基本执行步数，
关于步数的描述参见章节 3.8。
- ⑦..... “●”标志表示该指令可以有子处理过程。
关于子处理过程参见章节 3.5。
- ⑧..... 表示解释单条指令的页码。

2.3 顺序指令

2.3.1 触点指令

表 2.3 触点指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参见 章节
触点	LD		● 启动逻辑操作(启动触点逻辑操作)		* 1	● * 3	5-2
	LDI		● 启动 NOT 逻辑操作(启动 b 触点逻辑操作)				
	AND		● 逻辑乘(a 触点串行连接)				
	ANI		● 逻辑乘(b 触点串行连接)				
	OR		● 逻辑或(a 触点并行连接)				
	ORI		● 逻辑或非(b 触点并行连接)				
	LDP		● 启动前沿脉冲操作		* 2	● * 3	5-5
	LDF		● 启动后沿脉冲操作				
	ANDP		● 前沿脉冲串行连接				
	ANDF		● 后沿脉冲串行连接				
	ORP		● 前沿脉冲并行连接				
	ORF		● 后沿脉冲并行连接				

备注

1) * 1: 步数会由于所使用设备的不同而有变化。

软元件	步数
内部软元件, 文件寄存器(R0 到 R32767)	1
直接访问输入(DX)	2
上述软元件外的其它软元件	3

2) * 2: 步数会由于所使用的软元件的不同和 CPU 模块类型的不同而有变化。

软元件	步数	
	QCPU	QnACPU
内部软元件, 文件寄存器(R0 到 R32767)	1	2
直接访问输入(DX)	2	2
上述软元件外的其它软元件	3	3

3) * 3: 子处理过程只在 QCPU 上有效。

2.3.2 连接指令

表 2.4 连接指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
连接	ANB		● 逻辑块之间的 AND(逻辑块之间的串行连接)		1	—	5-7
	ORB		● 逻辑块之间的 OR(逻辑块之间的串行连接)				
	MPS		● 操作结果的存储器存储		1	—	5-9
	MRD		● 读 MPS 指令存储的操作结果				
	MPP		● 对 MPS 指令存储的操作结果的读和复位				
	INV		● 读 MPS 指令存储的操作结果		1	—	5-13
	MEP		● 操作结果的取反		1	—	5-14
	MEF		● 前沿脉冲操作结果的转换				
	EGP		● 后沿脉冲操作结果的转换		1	—	5-16
	EGF		● 前沿脉冲操作结果的转换(存储在 Vn)		* 1		

备注

* 1: 步数会由于所使用的 CPU 模块类型的不同而有变化。


部件	基本步数
高性能型 QCPU 过程 CPU QnACPU	1
基本型 QCPU	2

2.3.3 输出指令

表 2.5 输出指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
输出	OUT		● 软元件输出		* 1	—	5-18
	SET		● 设置软元件		* 1	—	5-28 5-32
	RST		● 复位软元件		* 1	—	5-30 5-32
	PLS		● 在输入信号的前沿处产生 1 个周期的程序脉冲		2	—	5-34
	PLF		● 在输入信号的后沿处产生 1 个周期的程序脉冲				
	FF		● 软元件输出的取反		2	—	5-36
	DELTA		● 直接输出的脉冲转化		2	—	5-38
	DELTAP						

备注

- 1) * 1: 步数会由于所使用软元件的不同而有变化。
关于步数参见单个指令的描述说明页。
- 2) * 2:  的执行条件只有在使用警报器 (F) 时才被应用。

2.3.4 移位指令

表 2.6 移位指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
移位	SFT		● 软元件的 1 位移位		2	—	5-40
	SFTP						

2.3.5 主控制指令

表 2.7 主控制指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子集	参见 章节
主站控制	MC		● 启动主控制		2	—	5-42
	MCR		● 复位主控制		1		

2.3.6 终止指令

表 2.8 终止指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子集	参见 章节
程序结束	FEND		● 主程序的终止		1	—	5-46
	END		● 顺序程序的终止				5-48

2.3.7 其它指令

表 2.9 其它指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子集	参见 章节
停止	STOP		<ul style="list-style-type: none"> ● 在输入条件得到满足后终止顺序程序 ● 将 RUN/STOP 键切换回到 RUN 位置后顺序程序重新被执行 		1	—	5-50
忽略	NOP	—	● 忽略(用于程序删除或空白区)		1	—	5-52
	NOPLF		● 忽略(在打印输出过程中改变页码号)				
	PAGE		● 忽略(并行程序从页码 n 的步骤 0 开始被控制)				

2.4 基本指令

2.4.1 比较操作指令

表 2.10 比较操作指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
16 位数据 比较	LD=		<ul style="list-style-type: none"> ● 当(S1) = (S2)时处于导通状态 ● 当(S1) ≠ (S2)时处于不导通状态 		3	●	6-2
	AND=						
	OR=						
	LD<>		<ul style="list-style-type: none"> ● 当(S1) ≠ (S2)时处于导通状态 ● 当(S1) = (S2)时处于不导通状态 		3	●	6-2
	AND<>						
	OR<>						
	LD>		<ul style="list-style-type: none"> ● 当(S1) > (S2)时处于导通状态 ● 当(S1) ≤ (S2)时处于不导通状态 		3	●	6-2
	AND>						
	OR>						
	LD<=		<ul style="list-style-type: none"> ● 当(S1) ≤ (S2)时处于导通状态 ● 当(S1) > (S2)时处于不导通状态 		3	●	6-2
	AND<=						
	OR<=						
	LD<		<ul style="list-style-type: none"> ● 当(S1) < (S2)时处于导通状态 ● 当(S1) ≥ (S2)时处于不导通状态 		3	●	6-2
	AND<						
	OR<						
	LD>=		<ul style="list-style-type: none"> ● 当(S1) ≥ (S2)时处于导通状态 ● 当(S1) < (S2)时处于不导通状态 		3	●	6-2
	AND>=						
	OR>=						

表 2.10 比较操作指令 (续表)

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
32 位数据 比较	LDD=		<ul style="list-style-type: none"> ● 当(S1+1, S1) = (S2+1, S2)时处于导通状态 ● 当(S1+1, S1) ≠ (S2+1, S2)时处于不导通状态 		*1	●	6-4
	ANDD=						
	ORD=						
	LDD<>		<ul style="list-style-type: none"> ● 当(S1+1, S1) ≠ (S2+1, S2)时处于导通状态 ● 当(S1+1, S1) = (S2+1, S2)时处于不导通状态 		*1	●	6-4
	ANDD<>						
	ORD<>						
	LDD>		<ul style="list-style-type: none"> ● 当(S1+1, S1) > (S2+1, S2)时处于导通状态 ● 态当(S1+1, S1) ≤ (S2+1, S2)时处于不导通状态 		*1	●	6-4
	ANDD>						
	ORD>						
	LDD≤		<ul style="list-style-type: none"> ● 当(S1+1, S1) ≤ (S2+1, S2)时处于导通状态 ● 当(S1+1, S1) > (S2+1, S2)时处于不导通状态 		*1	●	6-4
	ANDD≤						
	ORD≤						
	LDD<		<ul style="list-style-type: none"> ● 当(S1+1, S1) < (S2+1, S2)时处于导通状态 ● 当(S1+1, S1) ≥ (S2+1, S2)时处于不导通状态 		*1	●	6-4
	ANDD<						
	ORD<						
	LDD≥		<ul style="list-style-type: none"> ● 当(S1+1, S1) ≥ (S2+1, S2)时处于导通状态 ● 当(S1+1, S1) < (S2+1, S2)时处于不导通状态 		*1	●	6-4
	ANDD≥						
	ORD≥						

备注

※ 1: 执行步数可能随使用的软元件和 CPU 模块类型的不同而变化。

部件	基本步数	
高性能型 QCPU 过程 CPU	(1) 只有在使用下列设备时 ● 字软元件: 内部软元件(文件寄存器 ZR 除外) ● 位软元件: 软元件号是 16 的倍数, 其数字指定是 K8, 并且不使用索引修改的软元件 ● 常数: 无限制	注释 1) 5
	(2) 当使用(1)外的其它软元件	注释 2) 3
基本型 QCPU QnCPU	3	注释 2)

注释 1: 对于高性能型 QCPU 或过程 CPU, 在(1)中的执行步数更多但是执行速度也快。

注释 2: 执行步数会基于在章节 3.8 中描述的条件而增加。

表 2.10 比较操作指令(续表)

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
实数数据比较	LDE=		<ul style="list-style-type: none"> 当(S1+1, S1)=(S2+1, S2)时处于导通状态 当(S1+1, S1) ≠ (S2+1, S2)时处于不导通状态 		3	—	6-6
	ANDE=						
	ORE=						
	LDE<>		<ul style="list-style-type: none"> 当(S1+1, S1) ≠ (S2+1, S2)时处于导通状态 当(S1+1, S1) = (S2+1, S2)时处于不导通状态 		3	—	6-6
	ANDE<>						
	ORE<>						
	LDE>		<ul style="list-style-type: none"> 当(S1+1, S1) > (S2+1, S2)时处于导通状态 当(S1+1, S1) ≤ (S2+1, S2)时处于不导通状态 		3	—	6-6
	ANDE>						
	ORE>						
	LDE<=		<ul style="list-style-type: none"> 当(S1+1, S1) ≤ (S2+1, S2)时处于导通状态 当(S1+1, S1) > (S2+1, S2)时处于不导通状态 		3	—	6-6
	ANDE<=						
	ORE<=						
	LDE<		<ul style="list-style-type: none"> 当(S1+1, S1) < (S2+1, S2) 时处于导通状态 当(S1+1, S1) ≥ (S2+1, S2)时处于不导通状态 		3	—	6-6
	ANDE<						
	ORE<						
	LDE>=		<ul style="list-style-type: none"> 当(S1+1, S1) ≥ (S2+1, S2)时处于导通状态 当(S1+1, S1) < (S2+1, S2)时处于不导通状态 		3	—	6-6
	ANDE>=						
	ORE>=						

表 2.10 比较操作指令(续表)

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
字符串数据 比较	LD\$=		● 比较字符串 S1 和字符串 S2，一次一个字符。*		3	—	6-8
	AND\$=		● 态当(字符串 S1) = (字符串 S2)时处于导通状态				
	OR\$=		● 当(字符串 S1) ≠ (字符串 S2)时处于不导通状态				
	LD\$<>		● 比较字符串 S1 和字符串 S2，一次一个字符。*		3	—	6-8
	AND\$<>		● 当(字符串 S1) ≠ (字符串 S2)时处于导通状态				
	OR\$<>		● 当(字符串 S1) = (字符串 S2)时处于不导通状态				
	LD\$>		● 比较字符串 S1 和字符串 S2，一次一个字符。*		3	—	6-8
	AND\$>		● 当(字符串 S1) > (字符串 S2)时处于导通状态				
	OR\$>		● 当(字符串 S1) ≤ (字符串 S2)时处于不导通状态				
	LD\$<=		● 比较字符串 S1 和字符串 S2，一次一个字符。*		3	—	6-8
	AND\$<=		● 当(字符串 S1) ≤ (字符串 S2)时处于导通状态				
	OR\$<=		● 当(字符串 S1) > (字符串 S2)时处于不导通状态				
	LD\$<		● 比较字符串 S1 和字符串 S2，一次一个字符。*		3	—	6-8
	AND\$<		● 当(字符串 S1) < (字符串 S2)时处于导通状态				
	OR\$<		● 当(字符串 S1) ≥ (字符串 S2)时处于不导通状态				
	LD\$>=		● 比较字符串 S1 和字符串 S2，一次一个字符。*		3	—	6-8
	AND\$>=		● 当(字符串 S1) ≥ (字符串 S2)时处于导通状态				
	OR\$>=		● 当(字符串 S1) < (字符串 S2)时处于不导通状态				

备注

1) *：可以执行字符串比较的条件如下所示：





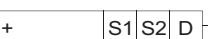



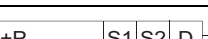

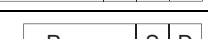

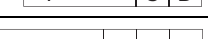

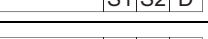





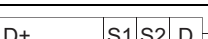

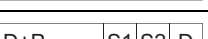

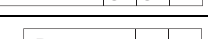

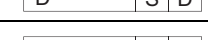

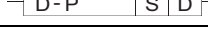

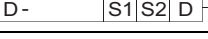

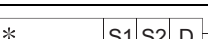

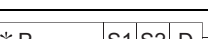

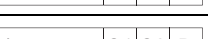

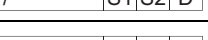



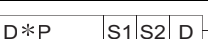

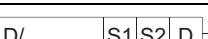

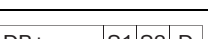

- 匹配： 串中的所有字节都匹配。
- 较大的字符串： 如果字符串不同，取字符代码数最大的字符串。如果字符串的长度不同，取最长的字符串。
- 较小的字符串： 如果字符串不同，取字符代码数最小的字符串。如果字符串的长度不同，取最短的字符串。

表 2.10 比较操作指令(续表)

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
块数据比较	BKCMP=		<ul style="list-style-type: none">在 1 字单元内从 S1 开始的 n 点数据和从 S2 开始的 n 点数据进行比较, 并且将比较结果存储到从(D)指定的位软元件的 n 个点。		5	—	6-12
	BKCMP<>						
	BKCMP>						
	BKCMP<=						
	BKCMP<						
	BKCMP>=						
	BKCMP=P						
	BKCMP< >P						
	BKCMP>P						
	BKCMP<=P						
	BKCMP<P						
	BKCMP>=P						

2.4.2 算术操作指令

表 2.11 算术操作指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子集	参见 章节
BIN16 位加 和减操作	+		● (D)+(S)→(D)		3	●	6-16
	+P						
	+		● (S1)+(S2)→(D)		4	●	6-18
	+P						
	-		● (D)-(S)→(D)		3	●	6-16
	-P						
	-		● (S1)-(S2)→(D)		4	●	6-18
	-P						
BIN32 位加 和减操作	D+		● (D+1, D)+(S+1, S)→(D+1, D)		*1	●	6-20
	D+P						
	D+		● (S1+1, S1)+(S2+1, S2)→(D+1, D)		*2	●	6-22
	D+P						
	D-		● (D+1, D)-(S+1, S)→(D+1, D)		*1	●	6-20
	D-P						
	D-		● (S1+1, S1)-(S2+1, S2)→(D+1, D)		*2	●	6-22
	D-P						
BIN16 位乘 法和除法操作	*		● (S1) * (S2)→(D+1, D)		*3	●	6-24
	*P						
	/		● (S1)/(S2)商(D), 余数(D+1)		4	●	6-24
	/P						
BIN32 位乘 法和除法操作	D*		● (S1+1, S1) * (S2+1, S2)→(D+3, D+2, D+1, D)		4	●	6-26
	D*P						
	D/		● (S1+1, S1)/(S2+1, S2)→商(D+3, D), 余数(D+3, D+2)		4	●	6-26
	D/P						

备注

1) * 1: 执行步数可能随使用的软元件和 CPU 模块类型的不同而变化。

部件	基本步数	
高性能型 QCPU 过程 CPU	(1) 只有在使用下列设备时 ● 字软元件: 内部软元件(文件寄存器 ZR 除外) ● 位软元件: 软元件号是 16 的倍数, 数字指定是 K8, 和使用无索引修改的软元件 ● 常数: 无限制	注释 1) 5
	(2) 当使用(1)外其它软元件时	注释 2) 3
基本型 QCPU QnCPU	3	注释 2)

注释 1: 对于高性能型 QCPU 或过程 CPU, 在(1)中的执行步数更多但是执行速度也快。

注释 2: 执行步数会基于在章节 3.8 中描述的条件而增加。

2) *2: 执行步数可能随使用的软元件和 CPU 模块类型的不同而变化。

部件	基本步数	
高性能型 QCPU 过程 CPU	(1) 只有在使用下列设备时 ● 字软元件: 内部软元件(文件寄存器 ZR 除外) ● 位软元件: 软元件号是 16 的倍数, 数字指定是 K8, 和使用无索引修改的软元件 ● 常数: 无限制	注释 1) 6
	(2) 当使用(1)外其它软元件时	注释 2) 4
基本型 QCPU QnCPU	4	注释 2)

注释 1: 对于高性能型 QCPU 或过程 CPU, 在(1)中的执行步数更多但是执行速度也快。

注释 2: 执行步数会基于在章节 3.8 中描述的条件而增加。

3) * 3: 注释 2: 执行步数会基于在章节 3.8 中描述的条件而增加。

部件	基本步数	
QCPU	● 字软元件: 内部软元件(文件寄存器 ZR 除外) ● 位软元件: 软元件号是 16 的倍数, 数字指定是 K8, 和使用无索引修改的软元件 ● 常数: 无限制	3
	使用除以上软元件外的其它软元件时	注释 1) 4
QnACPU	● 字软元件: 内部软元件(文件寄存器 ZR 除外) ● 位软元件: 软元件号是 16 的倍数, 数字指定是 K8, 和使用无索引修改的软元件 ● 常数: 无限制	4
	使用除以上软元件外的其它软元件时	注释 1) 4

注释 1: 执行步数会基于在章节 3.8 中描述的条件而增加。









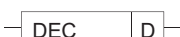



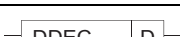

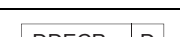

表 2.11 算术操作指令 (续表)

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
BCD4 位加 和减运算	B+		● $(D) + (S) \rightarrow (D)$		3	●	6-28
	B+P						
	B+		● $(S1) + (S2) \rightarrow (D)$		4	—	6-30
	B+P						
	B-		● $(D) - (S) \rightarrow (D)$		3	●	6-28
	B-P						
	B-		● $(S1) - (S2) \rightarrow (D)$		4	—	6-30
	B-P						
BCD8 位加 和减运算	DB+		● $(D+1, D) + (S+1, S) \rightarrow (D+1, D)$		3	—	6-32
	DB+P						
	DB+		● $(S1+1, S1) + (S2+1, S2) \rightarrow (D+1, D)$		4	—	6-34
	DB+P						
	DB-		● $(D+1, D) - (S+1, S) \rightarrow (D+1, D)$		3	—	6-32
	DB-P						
	DB-		● $(S1+1, S1) - (S2+1, S2) \rightarrow (D+1, D)$		4	—	6-34
	DB-P						
BCD4 位乘 和除运算	B*		● $(S1) * (S2) \rightarrow (D+1, D)$		4	●	6-36
	B*P						
	B/		● $(S1) / (S2) \rightarrow \text{商}(D), \text{余数}(D+1)$		4	●	6-36
	B/P						
BCD4 位乘 和除运算	DB*		● $(S1+1, S1) * (S2+1, S2) \rightarrow (D+3, D+2, D+1, D)$		4	—	6-38
	DB*P						
	DB/		● $(S1+1, S1) / (S2+1, S2) \rightarrow \text{商}(D+1, D), \text{余数}(D+3, D+2)$		4	●	6-38
	DB/P						

表 2.11 算术操作指令 (续表)

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
浮点十进制 数据加和减 运算	E+		● (D)+(S)→(D)		3	—	6-40
	E+P						
	E+		● (S1)+(S2)→(D)		4	—	6-42
	E+P						
	E-		● (D)-(S)→(D)		3	—	6-40
	E-P						
	E-		● (S1)-(S2)→(D)		4	—	6-42
	E-P						
浮点十进制 数据乘和除 运算	E*		● (D+1, D)+(S+1, S)→(D+1, D)		3	—	6-44
	E*P						
	E/		● (S1+1, S1)+(S2+1, S2)→(D+1, D)		4	—	6-44
	E/P						
BIN 块加和 减运算	BK+		● (D+1, D)-(S+1, S)→(D+1, D)		5	—	6-46
	BK+P						
	BK-		● (S1+1, S1)-(S2+1, S2)→(D+1, D)		5	—	6-46
	BK-P						
字符串数据 合并	\$+		● (S1)*(S2)→(D+1, D)		3	—	6-49
	\$+P						
	\$+		● (S1)/(S2)→商(D), 余数(D+1)		4	—	6-51
	\$+P						

表 2.11 算术操作指令 (续表)

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
BIN 数据增 加	INC		● (D)+1→(D)		2	●	6-53
	INCP						
	DINC		● (D+1, D)+1→(D+1, D)		* 1	●	6-55
	DINCP						
	DEC		● (D)-1→(D)		2	●	6-53
	DECP						
	DDEC		● (D+1, D)-1→(D+1, D)		* 1	●	6-55
	DDECP						

备注

1) * 1: 执行步数可能随使用的软元件和 CPU 模块类型的不同而变化

部件	基本步数	
高性能型 QCPU 过程 CPU	(1) 只有在使用下列设备时 ● 字软元件: 内部软元件(文件寄存器 ZR 除外) ● 位软元件: 软元件号是 16 的倍数, 数字指定是 K8, 和使用无索引修改的软元件 ● 常数: 无限制	注释 1) 3
	(2) 当使用(1)外其它软元件时	注释 2) 2
基本型 QCPU QnCPU	2	注释 2)

注释 1: 对于高性能型 QCPU 或过程 CPU, 在(1)中的执行步数更多但是执行速度也快。

注释 2: 执行步数会基于在章节 3.8 中描述的条件而增加。

2.4.3 数据转换指令

表 2.12 数据转换指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参见 章节
BCD 转换	BCD		BCD 转换 ● (S) → (D) BIN (0到9999)		3	●	6-57
	BCDP						
	DBC		BCD 转换 ● (S+1, S) → (D+1, D) BIN (0到99999999)		3	●	6-57
	DBCDP						
BIN 转换	BIN		BIN 转换 ● (S) → (D) BCD (到9999)		3	●	6-59
	BINP						
	DBIN		BIN 转换 ● (S+1, S) → (D+1, D) BCD (0到99999999)		3	●	6-59
	DBINP						
从 BIN 转换 成浮点十进 制数	FLT		转换成浮点十 进制数 ● (S+1, S) → (D+1, D) BIN (-32768到32767)		3	—	6-61
	FLTP						
	DFLT		转换成浮点 十进制数 ● (S+1, S) → (D) 实数 (-2147483648到2147483647)		3	—	6-61
	DFLTP						
从浮点十进 制数转换成 BIN	INT		转换成BIN ● (S+1, S) → (D) 实数 (-32768到32767)		3	—	6-63
	INTP						
	DINT		转换成BIN ● (S+1, S) → (D+1, D) 实数 (-2147483648到2147483647)		3	—	6-63
	DINTP						
BIN16 位和 32 位之间 的转换	DBL		转换 ● (S) → (D+1, D) BIN (-32768到32767)		3	—	6-65
	DBLP						
	WORD		转换 ● (S+1, S) → (D) BIN (-32768到32767)		3	—	6-66
	WORDP						
BIN 转换成 格雷码	GRY		转换成格雷码 ● (S) → (D) BIN (-32768到32767)		3	—	6-67
	GRYP						
	DGRY		转换成格雷码 ● (S+1, S) → (D+1, D) BIN (-2147483648到2147483647)		3	—	6-67
	DGRYP						

表 2.12 数据转换指令(续表)

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子集	参见 章节
格雷码转换 成 BIN	GBIN		● 转换成BIN数据 格雷码 (S) → (D) (-32768 to 32767)		3	—	6-69
	GBINP		● 转换成BIN数据 格雷码 (S) → (D) (-32768 to 32767)		3	—	6-69
	DGBIN		● (S+1, S) 转换成BIN数据 格雷码 → (D+1, D) (-2147483648 to 2147483647)		3	—	6-69
	DGBINP		● (S+1, S) 转换成BIN数据 格雷码 → (D+1, D) (-2147483648 to 2147483647)		3	—	6-69
补足到 2	NEG		● (D) 转换成BIN数据 BIN 数据 → (D)		2	—	6-71
	NEGP		● (D) 转换成BIN数据 BIN 数据 → (D)		2	—	6-71
	DNEG		● (D+1, D) 转换成BIN数据 BIN 数据 → (D+1, D)		2	—	6-71
	DNEGP		● (D+1, D) 转换成BIN数据 BIN 数据 → (D+1, D)		2	—	6-71
	ENEG		● (D+1, D) 转换成BIN数据 实型数据 → (D+1, D)		2	—	6-73
	ENEGP		● (D+1, D) 转换成BIN数据 实型数据 → (D+1, D)		2	—	6-73
块转换	BKBCD		● 将从(S)开始的 n 个数据点的 BIN 数据 批量转换成 BCD 数据, 并从(D)处向前 存储操作结果		4	—	6-74
	BKBCDP		● 将从(S)开始的 n 个数据点的 BIN 数据 批量转换成 BCD 数据, 并从(D)处向前 存储操作结果		4	—	6-74
	BKBIN		● 将从(S)开始的 n 个数据点的 BCD 数据 批量转换成 BIN 数据, 并从(D)处向前 存储操作结果		4	—	6-76
	BKBINP		● 将从(S)开始的 n 个数据点的 BCD 数据 批量转换成 BIN 数据, 并从(D)处向前 存储操作结果		4	—	6-76

2.4.4 数据传送指令

表 2.13 数据传送指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子集	参见 章节
16 位数据 传送	MOV		● (S) → (D)		*4	●	6-78
	MOVP				*1		
32 位数据 传送	DMOV		● (S+1, S) → (D+1, D)		*2	●	6-78
	DMOVP						
浮点十进制 数据传送	EMOV		● (S+1, S) → (D+1, D) 实型数据		*2	● *3	6-80
	EMOVP						
字符串数据 传送	\$MOV		● 将由(S)指定的字符串传送到由(D)指定的 软元件, 向前存储		3	—	6-82
	\$MOVP						
16 位负数 数据传送	CML		● (S) → (D)		*1	●	6-84
	CMLP						
32 位负数 数据传送	DCML		● (S+1, S) → (D+1, D)		*2	●	6-84
	DCMLP						
块传送	BMOV				4	●	6-87
	BMOVP						
同一数据块 的多路传送	FMOV				4	●	6-89
	FMOVP						
16 位数据 交换	XCH		● (S) ↔ (D)		3	●	6-91
	XCHP						
32 位数据 交换	DXCH		● (S+1, S) ↔ (D+1, D)		3	●	6-91
	DXCHP						
块数据交换	BXCH				4	—	6-93
	BXCHP						
高字节和低 字节间的交 换	SWAP				3	—	6-95
	SWAPP						

备注

1) * 1: 执行步数可能随使用的软元件和 CPU 模块类型的不同而变化。

部件	基本步数	
QCPU	(1) 只有在使用下列设备时 ● 字软元件: 内部软元件(文件寄存器 ZR 除外) ● 位软元件: 软元件号是 16 的倍数, 数字指定是 K8, 和使用无索引修改的软元件 ● 常数: 无限制	2
	(2) 当使用(1)外其它软元件时	注释 1) 3
QnACPU	3	注释 1)

注释 1: 执行步数会基于在章节 3.8 中描述的条件而增加。

2) * 2: 执行步数可能随使用的软元件和 CPU 模块类型的不同而变化。

部件	基本步数	
高性能型 QCPU 过程 CPU	(1) 只有在使用下列设备时 ● 字软元件: 内部软元件(文件寄存器 ZR 除外) ● 位软元件: 软元件号是 16 的倍数, 数字指定是 K8, 和使用无索引修改的软元件 ● 常数: 无限制	2
	(2) 当使用(1)以外其它软元件时	注释 1) 3
基本模型 QCPU	(1) 只有在使用下列设备时 ● 字软元件: 内部软元件(文件寄存器 ZR 除外) ● 位软元件: 软元件号是 16 的倍数, 数字指定是 K8, 和使用无索引修改的软元件 ● 常数: 无限制 (当使用上面的软元件+常量时, 实际步数为 3)	2
	(2) 当使用(1)以外其它软元件时	注释 1) 3
QnCPU	3	注释 1)

注释 1: 执行步数会基于在章节 3.8 中描述的条件而增加。

3) * 3: 只有在 QCPU 中子集才有效。

4) * 4: 执行步数可能随使用的软元件和 CPU 模块类型的不同而变化。

部件	基本步数	
QCPU, QnACPU	(1) 只有在使用下列设备时 ● 字软元件: 内部软元件(文件寄存器 ZR 除外) ● 位软元件: 软元件号是 16 的倍数, 数字指定是 K8, 和使用无索引修改的软元件 ● 常数: 无限制	3
	(2) 当使用(1)以外其它软元件时	注释 1)3

注释 1: 执行步数会基于在章节 3.8 中描述的条件而增加。

2.4.5 程序分支指令

表 2.14 程序分支指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
跳转	CJ		● 当输入条件满足时跳转到 Pn		2	●	6-96
	SCJ		● 在输入条件得到满足以后的下一个扫描周期跳转到 Pn		2	●	6-96
	JMP		● 无条件地跳转到 Pn		2	●	6-96
	GOEND		● 当输入条件得到满足时，跳转到 END 指令		1	—	6-99

2.4.6 程序执行控制指令

表 2.15 程序执行控制指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
使中断无效	DI		● 禁止中断程序的运行		1	—	6-100
使中断有效	EI		● 允许中断程序的执行		1	—	6-100
中断无效/ 有效设置	IMASK		● 对每个中断程序禁止或允许中断		2	—	6-100
返回	IRET		● 在中断程序之后返回到顺序程序		1	—	6-109

2.4.7 I/O 刷新指令

表 2.16 I/O 刷新指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
I/O 刷新	RFS		● 在扫描过程中刷新相关的 I/O 区域		3	—	6-111
	RFSP						

2.4.8 其它使用方便的指令

表 2.17 其它使用方便的指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
递增/递减 计数器	UDCNT1		<p>(S)+0</p> <p>(S)+1</p> <p>当前Cn值</p> <p>Cn接触点</p>		4	—	6-113
	UDCNT2		<p>(S)+0</p> <p>(S)+1</p> <p>当前Cn值</p> <p>Cn接触点</p>		4	—	6-115
教学定时器	TTMR		<p>● (TTMR为ON的时间) * n → (D)</p> <p>n = 0:1, n = 1:10, n = 2:100</p>		3	—	6-117
特殊定时器	STMR		<p>● 对从由(D)指定的位软元件开始的4点的操作如下所示,这取决于用于STMR指令的输入条件的ON/OFF状态:</p> <p>(D)+0: 关闭延迟定时器输出</p> <p>(D)+1: 关闭定时器输出后的一拍</p> <p>(D)+2: 开启定时器输出后的一拍</p> <p>(D)+3: 启延迟定时器输出</p>		3	—	6-119
最近路径控制	ROTC		<p>● 从停止位置到由(S+1)指定的位置,按照最短路径以n1个分区循环移动循环表。</p>		5	—	6-122
斜坡信号	RAMP		<p>● 从n1到n2,对由D1指定的软元件数据在n3个扫描周期内改变这些软元件数据。</p>		6	—	6-124
脉冲密度	SPD		<p>● 在由n指定的时间周期内计数从由(S)指定的软元件发出的脉冲输入,并且将计数结果存储到由(D)指定的软元件中。</p>		4	—	6-126
输出	PLSY		<p>● (n1)Hz → (D)</p> <p>输出 n2 次</p>		4	—	6-128
脉冲宽度调制	PWM		<p>(D)</p>		4	—	6-130
矩阵输入	MTR		<p>● 将由(S)指定的软元件中的n行,顺序存储到由(D2)指定的软元件中。</p>		5	—	6-132

2.5 应用指令

2.5.1 逻辑操作指令

表 2.18 逻辑操作指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
逻辑乘积	WAND		● $(D) \wedge (S) \rightarrow (D)$		3	●	7-3
	WANDP						
	WAND		● $(S1) \wedge (S2) \rightarrow (D)$		4	● *3	7-5
	WANDP						
	DAND		● $(D+1, D) \wedge (S+1, S) \rightarrow (D+1, D)$		*1	●	7-3
	DANDP						
	DAND		● $(S1+1, S1) \wedge (S2+1, S2) \rightarrow (D+1, D)$		*2	● *3	7-5
	DANDP						
	BKAND				5	—	7-8
	BKANDP						
逻辑求和	WOR		● $(D) \vee (S) \rightarrow (D)$		3	●	7-10
	WORP						
	WOR		● $(S1) \vee (S2) \rightarrow (D)$		4	● *3	7-12
	WORP						
	DOR		● $(D+1, D) \vee (S+1, S) \rightarrow (D+1, D)$		*1	●	7-10
	DORP						
	DOR		● $(S1+1, S1) \vee (S2+1, S2) \rightarrow (D+1, D)$		*2	● *3	7-12
	DORP						
	BKOR				5	—	7-14
	BKORP						
专有 OR	WXOR		● $(D) \nabla (S) \rightarrow (D)$		3	●	7-16
	WXORP						
	WXOR		● $(S1) \nabla (S2) \rightarrow (D)$		4	● *3	7-18
	WXORP						

表 2.18 辑操作指令 (续表)

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
专有 OR	DXOR		● $(D+1, D) \vee (S+1, S) \rightarrow (D+1, D)$		*1	●	7-16
	DXORP						
	DXOR		● $(S1+1, S1) \vee (S2+1, S2) \rightarrow (D+1, D)$		*2	● *3	7-18
	DXORP						
	BKXOR				5	—	7-20
	BKXORP						
非专有逻辑 求和	WXNR		● $\overline{(D)} \vee \overline{(S)} \rightarrow (D)$		3	●	7-22
	WXNRP						
	WXNR		● $\overline{(S1)} \vee \overline{(S2)} \rightarrow (D)$		4	● *3	7-25
	WXNRP						
	DXNR		● $\overline{(D+1, D)} \vee \overline{(S+1, S)} \rightarrow (D+1, D)$		*1	●	7-22
	DXNRP						
	DXNR		● $\overline{(S1+1, S1)} \vee \overline{(S2+1, S2)} \rightarrow (D+1, D)$		*2	● *3	7-25
	DXNRP						
	BKXNR				5	—	7-28
	BKXNRP						

备注

1)* 1: 执行步数可能随使用的软元件和 CPU 模块类型的不同而变化。

部件	基本步数	
高性能型 QCPU 过程 CPU	(1) 只有在使用下列软元件时 ● 字软元件: 内部软元件(文件寄存器 ZR 除外) ● 位软元件: 软元件号是 16 的倍数, 数字指定是 K8, 和使用无索引修改的软元件 ● 常数: 无限制	注释 1) 5
	(2) 当使用(1)外其他软元件时	注释 2) 3
基本型 QCPU QnCPU	3	注释 2)

注释 1: 对于高性能型 QCPU 或过程 CPU, 在(1)中的执行步数更多但是执行速度也快。

注释 2: 执行步数会基于在章节 3.8 中描述的条件而增加。

2)* 2: 执行步数可能随使用的软元件和 CPU 模块类型的不同而变化。

部件	基本步数	
高性能型 QCPU 过程 CPU	(1) 只有在使用下列软元件时 ● 字软元件: 内部软元件(文件寄存器 ZR 除外) ● 位软元件: 软元件号是 16 的倍数, 数字指定是 K8, 和使用无索引修改的软元件 ● 常数: 无限制	注释 1) 6
	(2) 当使用(1)外其他软元件时	注释 2) 4
基本型 QCPU QnCPU	4	注释 2)

注释 1: 对于高性能型 QCPU 或过程 CPU, 在(1)中的执行步数更多但是执行速度也快。

注释 2: 执行步数会基于在章节 3.8 中描述的条件而增加。

3)* 3: 子集只对 QCPU 有效。

2.5.2 循环指令

表 2.19 循环指令

类别	指令符号	符号	详细处理过程	执行条件	基本步数	子集	参见章节
向右循环移位	ROR				3	●	7-30
	RORP				3	●	7-30
	RCR				3	●	7-30
	RCRP				3	●	7-30
向左循环移位	ROL				3	●	7-32
	ROLP				3	●	7-32
	RCL				3	●	7-32
	RCLP				3	●	7-32
向右循环移位	DROR				3	●	7-34
	DRORP				3	●	7-34
	DRCR				3	●	7-34
	DRCRP				3	●	7-34
向左循环移位	DROL				3	●	7-36
	DROLP				3	●	7-36
	DRCL				3	●	7-36
	DRCLP				3	●	7-36

2.5.3 位指令

表 2.20 移位指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子集	参见 章节
n 位移位	SFR				3	●	7-38
	SFRP						
	SFL				3	●	7-38
	SFLP						
1 位移位	BSFR				3	—	7-40
	BSFRP						
	BSFL				3	—	7-40
	BSFLP						
1-字移位	DSFR				3	●	7-42
	DSFRP						
	DSFL				3	●	7-42
	DSFLP						

2.5.4 处理指令

表 2.21 位处理指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子集	参见 章节
位置位/复 位	BSET				3	●	7-44
	BSETP						
	BRST				3	●	7-44
	BRSTP						

表 2.21 处理指令 (续表)

类别	指令符号	符号	详细处理过程	执行条件	基本步数	子集	参见章节
位测试	TEST				4	—	7-46
	TESTP						
	DTEST				4	—	7-46
	DTESTP						
位软元件批量复位	BKRST				3	—	7-48
	BKRSTP						

2.5.5 数据处理指令

表 2.22 数据处理指令

类别	指令符号	符号	详细处理过程	执行条件	基本步数	子集	参见章节
数据查找	SER				5	—	7-50
	SERP						
	DSER				5	—	7-50
	DSERP						
位检测	SUM				3	●	7-54
	SUMP						
	DSUM				3	●	7-54
	DSUMP						
解码	DECO				4	—	7-56
	DECOP						
编码	ENCO				4	—	7-58
	ENCOP						

表 2.22 数据处理指令 (续表)

类别	指令符号	符号	详细处理过程	执行条件	基本步数	子集	参见章节
7 段解码	SEG				3	●	7-60
	SEGP						
分解和链接	DIS		● 将由(S)指定的 16 位数据分解成 4 位单元, 并将结果存储到从(D)开始的 n 个数据点的低 4 位中。(n ≤ 4)		4	—	7-62
	DISP						
	UNI		● 链接由(S)指定的软元件的 n 个数据点的低 4 位, 并将结果存储到由(D)指定的软元件中。(n ≤ 4)		4	—	7-64
	UNIP						
	NDIS		● 将(S1)指定的软元件以后的软元件中的数据分解到(S2)指定的软元件以后的位中, 并按顺序存储到由(D)指定的软元件后的软元件中		4	—	7-66
	NDISP						
	NUNI		● 将(S1)指定的软元件以后的软元件中的数据链接到(S2)指定的软元件以后的位中, 并按顺序存储到由(D)指定的软元件后的软元件中		4	—	7-71
	NUNIP						
	WTOB		● 将由(S)指定的软元件中的 16 位数据分解成 8 位数据单元, 并按顺序存储到由(D)指定的软元件中。		4	—	7-71
	WTOBP						
	BTOW		● 将由(S)指定的软元件开始的 n 个数据点的 16 位数据的低 8 位数据链接成 16 位数据单元, 并按顺序存储到由(D)指定的软元件中。		4	—	7-71
	BTOWP						
查找	MAX		● 以 16 位数据为单元, 对由(S)指定的软元件开始的 n 个数据点进行数据搜索, 并将最大值存储到由(D)指定软元件中。		4	—	7-75
	MAXP						
	MIN		● 以 16 位数据为单元, 对由(S)指定的软元件开始的 n 个数据点进行数据搜索, 并将最小值存储到由(D)指定软元件中。		4	—	7-77
	MINP						
	DMAX		● 以 32 位数据为单元, 对由(S)指定的软元件开始的 2n 个数据点进行数据搜索, 并将最大值存储到由(D)指定软元件中。		4	—	7-75
	DMAXP						
	DMIN		● 以 32 位为数据单元, 对由(S)指定的软元件开始的 2n 个数据点进行数据搜索, 并将最小值存储到由(D)指定软元件中。		4	—	7-77
	DMINP						

表 2.22 数据处理指令(续表)

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
排序	SORT	<div><div><div>SORT</div><div>S1</div><div>n</div><div>S2</div><div>D1</div><div>D2</div></div></div> <div>● S2: 在一个运行中进行的比较数 ● D1在排序完成后变为ON的设备软元件 ● D2: 用于系统使用</div>	● 以 16 位数据为单元, 对由(S1)指定的软元件开始的 n 个数据点中的数据进行排序。(需要 $nx(n1)/2$ 个扫描周期)		6	—	7-80
	DSORT	<div><div><div>DSORT</div><div>S1</div><div>n</div><div>S2</div><div>D1</div><div>D2</div></div></div> <div>● S2: 在一个运行中进行的比较数 ● D1: 在排序完成后变为ON的设备元件 ● D2: 用于系统使用</div>	● 以 32 位数据为单元, 对由(S1)指定的软元件开始的 $2 * n$ 个数据点中的数据进行排序。(需要 $nx(n+1)/2$ 个扫描周期)				
总值计算	WSUM	<div><div><div>WSUM</div><div>S</div><div>D</div><div>n</div></div></div>	● 对由(S)指定的软元件开始的 n 个数据点的 16 位 BIN 数据进行加操作, 并存储到由(D)指定的软元件中。		4	—	7-83
	WSUMP	<div><div><div>WSUMP</div><div>S</div><div>D</div><div>n</div></div></div>	● 对由(S)指定的软元件开始的 n 个数据点的 16 位 BIN 数据进行加操作, 并存储到由(D)指定的软元件中。				
	DWSUM	<div><div><div>DWSUM</div><div>S</div><div>D</div><div>n</div></div></div>	● 对由(S)指定的软元件开始的 n 个数据点的 32 位 BIN 数据进行加操作, 并存储到由(D)指定的软元件中。				
	DWSUMP	<div><div><div>DWSUMP</div><div>S</div><div>D</div><div>n</div></div></div>	● 对由(S)指定的软元件开始的 n 个数据点的 32 位 BIN 数据进行加操作, 并存储到由(D)指定的软元件中。				7-85

2.5.6 结构体创建指令

表 2.23 结构体创建指令

类别	指令符号	符号	详细处理过程	执行条件	基本步数	子集	参见章节
重复数	FOR		● 在 FOR 和 NEXT 之间执行 n 次		2	—	7-87
	NEXT				1	—	
	BREAK		● 强制结束从 FOR 到 FOR 的周期循环执行并跳转指针到 Pn。		3	—	7-89
	BREAKP						
子程序调用	CALL		● 当输入条件满足时执行子程序 Pn(S1 到 Sn 是送往子程序的参数。0 ≤ n ≤ 5)		*1 2 + n	—	7-91
	CALLP						
	RET		● 从子程序返回		1	—	7-95
	FCALL		● 如果输入条件得不到满足则不执行子程序 Pn 的操作处理。(S1 到 Sn 是送往子程序的参数。0 ≤ n ≤ 5)		*1 2 + n	—	7-96
	FCALLP						
	ECALL	 *: 程序名称	● 当输入条件得到满足时执行指定的程序名内的子程序 Pn。(S1 到 Sn 是送往子程序的参数。0 ≤ n ≤ 5)		*2 3 + n	—	7-100
	ECALLP	 *: 程序名称					
	EFCALL	 *: 程序名称	● 如果输入条件得不到满足则不执行指定程序名称内的子程序 Pn 的操作处理。(S1 到 Sn 是送往子程序的参数。0 ≤ n ≤ 5)		*2 3 + n	—	7-104
	EFCALLP	 *: 程序名称					
	XCALL	 *: 程序名称	● 当输入条件得到满足时执行子程 Pn。 ● 如果输入条件得不到满足则不执行子程序 Pn 的操作处理。(S1 到 Sn 是送往子程序的参数。0 ≤ n ≤ 5)		*1 2 + n	—	7-108
	COM		● 执行链接刷新和一般数据处理。		1	—	7-112

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参见 章节
固定索引修改	IX		● 对每个用于软元件修改梯形图中的软元件进行索引修改		2	—	7-118
	IXEND				1	—	
	IXDEV		● 在由(D)指定的软元件后的软元件中, 存储在 IX 和 IXEND 之间执行的索引修改的修改值。		1	—	7-126
	IXSET				3	—	

- * 1: n 表示用于子程序的变量数
- * 2: n 表示用于子程序中的变量总数, 和程序名称步数。程序名称步数被计算为“程序中的字符数/2(十进制余数被舍入)”

2.5.7 数据表操作指令

表 2.24 数据表操作指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参见 章节
数据表处理	FIFW				3	—	7-131
	FIFWP						
	FIFR				3	—	7-133
	FIFRP						
	FPOP				3	—	7-135
	FPOPP						
	FINS				4	—	7-137
	FINSP						
	FDEL				4	—	7-137
	FDELP						




2.5.8 缓冲存储区访问指令

表 2.25 缓冲存储区访问指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
数据读取	FROM		● 从智能功能模块/特殊功能模块中以 16 位数据为单元读取数据		5	—	7-140
	FROMP						
	DFRO		● 从智能功能模块/特殊功能模块中以 32 位数据为单元读取数据		5	—	7-140
	DFROP						
数据写	TO		● 以 16 位数据为单元写数据到智能功能模块/特殊功能模块		5	—	7-143
	TOP						
	DTO		● 以 32 位数据为单元写数据到智能功能模块/特殊功能模块		5	—	7-143
	DTOP						



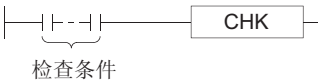















2.5.9 显示指令

表 2.26 显示指令

类别	指令 符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
ASCII 打印	PR	* SM701 为 OFF 时 PR S D		3	—	7-146
	PR	* SM701 为 OFF 时 PR S D				
	PRC	PRC S D			—	7-149
显示	LED	LED S		2	—	7-154
	LEDC	LEDC S				7-156
复位	LEDR	LEDR		1	—	7-158

2.5.10 调试和故障诊断指令

表 2.27 调试和故障诊断指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参见 章节
检查	CHKST		<ul style="list-style-type: none"> 当 CHKST 可执行时执行 CHK 指令 当 CHKST 处于不可执行状态时，跳至 CHK 指令之后的步。 		1	—	7-161
	CHK		<ul style="list-style-type: none"> 在正常条件下→SM80: OFF, SD80: 0 在不正常条件下→SM80: ON, SD80: Failure No. 				
	CHKCIR		<ul style="list-style-type: none"> 在由 CHK 指令检查的梯形图模式中开始更新 		1	—	7-165
	CHKEND		<ul style="list-style-type: none"> 在由 CHK 指令检查的梯形图模式中结束更新 				
状态闭锁	SLT		<ul style="list-style-type: none"> 执行状态闭锁 		1	—	7-173
	SLTR		<ul style="list-style-type: none"> 复位状态闭锁以使能重新执行 应用触发器到采样跟踪中 				
采样跟踪	STRA		<ul style="list-style-type: none"> 应用触发器到采样跟踪中 		1	—	7-175
	STRAR		<ul style="list-style-type: none"> 复位采样跟踪以使能重新执行 				
程序跟踪	PTRA		<ul style="list-style-type: none"> 应用触发器到程序跟踪 		1	—	7-177
	PTRAR		<ul style="list-style-type: none"> 复位程序跟踪以使能重新执行 				
	PTRAEXE		<ul style="list-style-type: none"> 执行程序跟踪 		1	—	7-177
	PTRAEXEP						

2.5.11 字符串处理指令

表 2.28 字符串处理指令































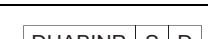



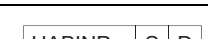

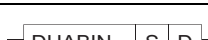

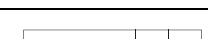

类别	指令符号	符号	详细处理过程	执行条件	基本步数	子集	参见章节
BIN 到十进制 ASCII	BINDA		● 将由(S)指定的 1 字 BIN 值转换成一个 5-位数的十进制 ASCII 值, 并将它存储到由(D)指定的字软元件中		3	—	7-179
	BINDAP						
	DBINDA		● 将由(S)指定的 2 字 BIN 值转换成一个 10-位数的十进制 ASCII 值, 并将它存储到由(D)指定号码的字软元件后的字软元件单元中		3	—	7-179
	DBINDAP						
BIN 到十六进制 ASCII	BINHA		● 将由(S)指定的 1 字 BIN 值转换成一个 4-位数的十六进制 ASCII 值, 并将它存储到由(D)指定的字软元件号码后的字软元件单元中		3	—	7-182
	BINHAP						
	DBINHA		● 将由(S)指定的 2 字 BIN 值转换成一个 8-位数的十六进制 ASCII 值, 并将它存储到由(D)指定号码的字软元件后的字软元件单元中		3	—	7-182
	DBINHAP						
BCD 到十进制 ASCII	BCDDA		● 将由(S)指定的 1 字 BCD 值转换成一个 4-位数的十进制 ASCII 值, 并将它存储到由(D)指定号码的字软元件后的字软元件单元中		3	—	7-185
	BCDDAP						
	DBCDDA		● 将由(S)指定的 2 字 BCD 值转换成一个 8-位数的十进制 ASCII 值, 并将它存储到由(D)指定号码的字软元件后的字软元件单元中		3	—	7-185
	DBCDDAP						
十进 ASCII 到 BIN	DABIN		● 将由(S)指定的 5-位数的十进制 ASCII 值转换成一个 1 字 BIN 值, 并将它存储到由(D)指定号码的字软元件中		3	—	7-188
	DABINP						
	DDABIN		● 将由(S)指定的 10-位数的十进制 ASCII 值转换成一个 2 字 BIN 值, 并将它存储到由(D)指定号码的字软元件中		3	—	7-188
	DDABINP						
十六进制 ASCII 到 BIN	HABIN		● (S)指定的 4-位数的十六进制 ASCII 值转换成一个 1 字 BIN 值, 并将它存储到由(D)指定号码的字软元件中		3	—	7-191
	HABINP						
	DHABIN		● 由(S)指定的 8-位数的十六进制 ASCII 值转换成一个 2 字 BIN 值, 并将它存储到由(D)指定号码的字软元件中		3	—	7-191
	DHABINP						

表 2.28 字符串处理指令(续表)

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
十进制 ASCII 到 BCD	DABCD		● 将由(S)指定的一个 4 位的十进制 ASCII 值转换成 1 字 BCD 值, 并将它存储到由 (D) 指定的字软元件中		3	—	7-193
	DABCDP						
	DDABCD		● 将由(S)指定的一个 8 位的十进制 ASCII 值转换成 2 字 BIN 值, 并将它存储到由 (D) 指定号码的字软元件后的字软元件单元中		3	—	7-193
	DDABCDP						
软元件注释 读操作	COMRD		● 将从(S)指定的软元件开始地注释存储到由 (D) 指定的软元件中		3	—	7-196
	COMRDP						
字符串长度 检测	LEN		● 将从(S)指定的字符串中的数据长度(字节数), 存储到由(D)指定的软元件中		3	—	7-201
	LENP						
BIN 到十进 制字符串	STR		● 将由(S2)指定的 1 字 BIN 值, 按照(S1)指定的总位数和小数点位数, 转换成十进制字符串, 并将它们存储到由 (D) 指定号码的软元件中		4	—	7-203
	STRP						
	DSTR		● 将由(S2)指定的 2 字 BIN 值, 按照(S1)指定的总位数和小数部分的位数, 转换成十进制字符串, 并将它们存储到由 (D) 指定号码的软元件中		4	—	7-203
	DSTRP						
十进制字 符串到 BIN	VAL		● 将由(S)指定的包含小数点的字符串, 转换成 1 字 BIN 值和小数部分的位数, 并将它们存储到由 (D1) 和 (D2) 指定号码的软元件中		4	—	7-209
	VALP						
	DVAL		● 将由(S)指定的包含小数点的字符串, 转换成 2 字 BIN 值和小数部分的位数, 并将它们存储到由 (D1) 和 (D2) 指定号码的软元件中		4	—	7-209
	DVALP						
浮点十进制 数到字符串	ESTR		● 将由(S1)指定的浮点十进制数据转换成字符串, 并将它们存储到由 (D) 指定号码的软元件中		4	—	7-214
	ESTRP						
字符串到浮 点十进制数	EVAL		● 将由(S)指定的字符串转换成浮点十进制数据, 并将它们存储到由 (D) 指定号码的软元件中		3	—	7-221
	EVALP						

表 2.28 字符串处理指令(续表)

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
十六进制 BIN 到 ASCII	ASC		● 将由(S)指定号码的软元件及其之后的软元件中的 1 字 BIN 值转换成 ASCII, 并只存储它们中的 n 个字符到由(D)指定号码的软元件中。		4	—	7-225
	ASCP						
ASCII 到十 六进制 BIN	HEX		● 只将由(S)指定号码的软元件及其之后的软元件中的 n 个 ASCII 字符转换成 BIN 值, 并将它们存储到由(D)指定号码的软元件中。		4	—	7-227
	HEXP						
字符串处理	RIGHT		● 将从(S)指定的字符串的最后开始的 n 个字符存储到由(D)指定的软元件中。		4	—	7-229
	RIGHTP						
	LEFT		● 将从(S)指定的字符串的开始的 n 个字符存储到由(D)指定的软元件中。		4	—	7-232
	LEFTP						
	MIDR		● 将在(S1)指定的字符串中从(S2)指定的位置开始的指定数量的字符存储到由(D)指定的软元件中。		4	—	7-232
	MIDRP						
	MIDW		● 将在(S1)指定的字符串中从(S2)指定的位置开始的指定数量的字符存储到由(D)指定的软元件中。		4	—	7-232
	MIDWP						
	INSTR		● 从字符串(S2)的第 n 个字符开始搜索字符串(S1), 并存储匹配的位置到(D)。		5	—	7-236
	INSTRP						
浮点十进制 数到 BCD	EMOD		● 按照由(S2)指定的十进制小数部分的位数将浮点十进制小数数据(S1)转换成 BCD 数据, 并存储到由(D)指定软元件中。		4	—	7-238
	EMODP						
BCD 到浮点 十进制数据	EREXP		● 按照由(S2)指定的十进制小数部分的位数将 BCD 数据(S1)转换成浮点十进制小数数据, 并存储到由(D)指定的软元件中。		4	—	7-240
	EREXPP						

2.5.12 特殊功能指令

表 2.29 特殊功能指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子集	参见 章节
三角函数功能 (浮点十进制小数数据)	SIN		● $\sin(S+1, S) \rightarrow (D+1, D)$		3	—	7-242
	SINP						
	COS		● $\cos(S+1, S) \rightarrow (D+1, D)$		3	—	7-244
	COSP						
	TAN		● $\tan(S+1, S) \rightarrow (D+1, D)$		3	—	7-246
	TANP						
	ASIN		● $\sin^{-1}(S+1, S) \rightarrow (D+1, D)$		3	—	7-248
	ASINP						
	ACOS		● $\cos^{-1}(S+1, S) \rightarrow (D+1, D)$		3	—	7-250
	ACOSP						
	ATAN		● $\tan^{-1}(S+1, S) \rightarrow (D+1, D)$		3	—	7-252
	ATANP						
在角度和弧度间的转换	RAD		● $(S+1, S) \rightarrow (D+1, D)$ 将角度转换成弧度		3	—	7-254
	RADP						
	DEG		● $(S+1, S) \rightarrow (D+1, D)$ 将弧度转换成角度		3	—	7-256
	DEGP						
方根	SQR		● $\sqrt{(S+1, S)} \rightarrow (D+1, D)$		3	—	7-258
	SQRP						
指数操作	EXP		● $e^{(S+1, S)} \rightarrow (D+1, D)$		3	—	7-260
	EXPP						
自然对数	LOG		● $\log_e(S+1, S) \rightarrow (D+1, D)$		3	—	7-262
	LOGP						
随机数的产生	RND		● 产生一个随机数(从 0 到小于 32767)并将它存储到由(D)指定的软元件中。		2	—	7-264
	RNDP						
随机数序列的更新	SRND		● 按照存储在由(S)指定的软元件中的16-位 BIN 数据更新随机数序列。				
	SRNDP						

表 2.29 特殊功能指令(续表)

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
方根	BSQR		● $\sqrt{(S)} \rightarrow (D)+0$ +1 整数部分 小数部分		3	—	7-266
	BSQRP						
	BDSQR		● $\sqrt{(S+1, S)} \rightarrow (D)+0$ +1 整数部分 小数部分		3	—	7-266
	BDSQRP						
三角函数功 能	BSIN		● $\sin(S) \rightarrow (D)+0$ +1 正负号 整数部分 +2 小数部分		3	—	7-269
	BSINP						
	BCOS		● $\cos(S) \rightarrow (D)+0$ +1 正负号 整数部分 +2 小数部分		3	—	7-271
	BCOSP						
	BTAN		● $\tan(S) \rightarrow (D)+0$ +1 正负号 整数部分 +2 小数部分		3	—	7-273
	BTANP						
	BASIN		● $\sin^{-1}(S) \rightarrow (D)+0$ +1 正负号 整数部分 +2 小数部分		3	—	7-275
	BASINP						
	BACOS		● $\cos^{-1}(S) \rightarrow (D)+0$ +1 正负号 整数部分 +2 小数部分		3	—	7-277
	BACOSP						
	BATAN		● $\tan^{-1}(S) \rightarrow (D)+0$ +1 正负号 整数部分 +2 小数部分		3	—	7-279
	BATANP						









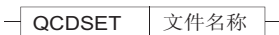



2.5.13 数据控制指令

表 2.30 数据控制指令

类别	指令符号	符号	详细处理过程	执行条件	基本步数	子集	参见章节
最高和最低限控制	LIMIT		<ul style="list-style-type: none"> 当 $(S3) < (S1)$ 时 存储 $(S1)$ 的值到 (D) 当 $(S1) \leq (S3) \leq (S2)$ 存储 $(S3)$ 的值到 (D) 当 $(S2) < (S3)$ 存储 $(S2)$ 的值到 (D) 		5	—	7-281
	LIMITP		<ul style="list-style-type: none"> 当 $(S3) < (S1)$ 时 存储 $(S1)$ 的值到 (D) 当 $(S1) \leq (S3) \leq (S2)$ 存储 $(S3)$ 的值到 (D) 当 $(S2) < (S3)$ 存储 $(S2)$ 的值到 (D) 		5	—	7-281
	DLIMIT		<ul style="list-style-type: none"> 当 $((S3)+1, (S3)) < ((S1)+1, (S1))$ 时 ... 存储 $((S1)+1, (S1))$ 的值到 $((D)+1, (D))$ 当 $((S1)+1, (S1)) \leq ((S3)+1, (S3)) < (S2+1, S2)$... 存储 $((S3)+1, (S3))$ 的值到 $((D)+1, (D))$ 当 $((S2), (S2)+1) < ((S3), (S3)+1)$ 时 ... 存储 $((S2)+1, (S2))$ 的值到 $((D)+1, (D))$ 		5	—	7-281
	DLIMITP		<ul style="list-style-type: none"> 当 $((S3)+1, (S3)) < ((S1)+1, (S1))$ 时 ... 存储 $((S1)+1, (S1))$ 的值到 $((D)+1, (D))$ 当 $((S1)+1, (S1)) \leq ((S3)+1, (S3)) < (S2+1, S2)$... 存储 $((S3)+1, (S3))$ 的值到 $((D)+1, (D))$ 当 $((S2), (S2)+1) < ((S3), (S3)+1)$ 时 ... 存储 $((S2)+1, (S2))$ 的值到 $((D)+1, (D))$ 		5	—	7-281
死区控制	BAND		<ul style="list-style-type: none"> 当 $(S1) \leq (S3) \leq (S2)$ $0 \rightarrow (D)$ 当 $(S3) < (S1)$, $(S3)-(S1) \rightarrow (D)$ 当 $(S2) < (S3)$ $S3-(S2) \rightarrow (D)$ 		5	—	7-284
	BANDP		<ul style="list-style-type: none"> 当 $(S1) \leq (S3) \leq (S2)$ $0 \rightarrow (D)$ 当 $(S3) < (S1)$, $(S3)-(S1) \rightarrow (D)$ 当 $(S2) < (S3)$ $S3-(S2) \rightarrow (D)$ 		5	—	7-284
	DBAND		<ul style="list-style-type: none"> 当 $((S1)+1, (S1)) \leq ((S3)+1, (S3)) \leq (S2)+1, (S2)$... $0 \rightarrow ((D)+1, (D))$ 当 $((S3)+1, (S3)) < ((S1)+1, (S1))$... $((S3)+1, (S3)) - ((S1)+1, (S1)) \rightarrow ((D)+1, (D))$ 当 $((S2)+1, (S2)) < ((S3)+1, (S3))$... $((S3)+1, (S3)) - ((S2)+1, (S2)) \rightarrow ((D)+1, (D))$ 		5	—	7-284
	DBANDP		<ul style="list-style-type: none"> 当 $((S1)+1, (S1)) \leq ((S3)+1, (S3)) \leq (S2)+1, (S2)$... $0 \rightarrow ((D)+1, (D))$ 当 $((S3)+1, (S3)) < ((S1)+1, (S1))$... $((S3)+1, (S3)) - ((S1)+1, (S1)) \rightarrow ((D)+1, (D))$ 当 $((S2)+1, (S2)) < ((S3)+1, (S3))$... $((S3)+1, (S3)) - ((S2)+1, (S2)) \rightarrow ((D)+1, (D))$ 		5	—	7-284
区域控制	ZONE		<ul style="list-style-type: none"> 当 $(S3) = 0$ $0 \rightarrow (D)$ 当 $(S3) > 0$ $(S3)+(S2) \rightarrow (D)$ 当 $(S3) < 0$ $(S3)-(S1) \rightarrow (D)$ 		5	—	7-287
	ZONEP		<ul style="list-style-type: none"> 当 $(S3) = 0$ $0 \rightarrow (D)$ 当 $(S3) > 0$ $(S3)+(S2) \rightarrow (D)$ 当 $(S3) < 0$ $(S3)-(S1) \rightarrow (D)$ 		5	—	7-287
	DZONE		<ul style="list-style-type: none"> 当 $((S3)+1, (S3)) = 0$... $0 \rightarrow ((D)+1, (D))$ 当 $((S3)+1, (S3)) > 0$... $((S3)+1, (S3)) + ((S2)+1, (S2)) \rightarrow ((D)+1, (D))$ 当 $((S3)+1, (S3)) < 0$... $((S3)+1, (S3)) + ((S1)+1, (S1)) \rightarrow ((D)+1, (D))$ 		5	—	7-287
	DZONEP		<ul style="list-style-type: none"> 当 $((S3)+1, (S3)) = 0$... $0 \rightarrow ((D)+1, (D))$ 当 $((S3)+1, (S3)) > 0$... $((S3)+1, (S3)) + ((S2)+1, (S2)) \rightarrow ((D)+1, (D))$ 当 $((S3)+1, (S3)) < 0$... $((S3)+1, (S3)) + ((S1)+1, (S1)) \rightarrow ((D)+1, (D))$ 		5	—	7-287

2.5.14 转换指令

表 2.31 转换指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
块号码标定	RSET		● 将扩展文件寄存器块的号码转换成由(S)指定的号码。		2	—	7-290
	RSETP						
文件设定	QDRSET		● 将文件名称用作文件寄存器		* 2 + n	—	7-292
	QDRSETP						
	QCDSET		● 将文件名称用作注释文件		* 2 + n	—	7-294
	QCDSETP						

* n([文件名称字符数]/2)表示一步。(小数部分舍入)





2.5.15 时钟指令

表 2.32 时钟指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子集	参见 章节
读/写时钟 数据	DATERD		● (时钟设备元件) → (D)+0 +1 年 +2 月 +3 日 +4 小时 +5 分 +6 秒		2	—	7-296
	DATERDP						
	DATEWR		● (D)+0 年 → (时钟设备元件) +1 月 +2 日 +3 小时 +4 分钟 +5 秒 +6 星期		2	—	7-300
	DATEWRP						
时钟数据的 加/减	DATE+		(S1) 小时 分钟 秒 + (S2) 小时 分钟 秒 → (D) 小时 分钟 秒		4	—	7-304
	DATE+P						
	DATE-		(S1) 小时 分 秒 - (S2) 小时 分 秒 → (D) 小时 分 秒		4	—	7-306
	DATE-P						
时钟数据传 送	SECOND		(S) 小时 分钟 秒 → (D) 秒 (低级1) 秒 (高级1)		3	—	7-308
	SECONDP						
	HOUR		(S) 秒 (低级) 秒 (高级) → (D) 小时 分钟 秒				
	HOURP						

2.5.16 外围软元件指令

表 2.33 外围软元件指令

类别	指令符号	符号	详细处理过程	执行条件	基本步数	子集	参见章节
输入/输出到外围软元件	MSG		● 将由(S)指定的消息存储到 QnACPU 此消息显示到外围软元件上		2	—	7-310
	PKEY		● 从外围软元件中输入的数据存储到由(D)指定的软元件上。		2	—	7-312

2.5.17 程序控制指令

表 2.34 程序控制指令

类别	指令符号	符号	详细处理过程	执行条件	基本步数	子集	参见章节
程序控制指令	PSTOP		● 将指定的程序置为备用状态		* 2 + n	—	7-315
	PSTOPP						
	POFF		● 将指定程序的 OUT 指令线圈置为 OFF, 并将指定的程序置为备用状态		* 2 + n	—	7-316
	POFFP						
	PSCAN		● 将指定程序登记为扫描执行类型		* 2 + n	—	7-318
	PSCANP						
	PLOW		● 将指定程序登记为低速执行类型		* 2 + n	—	7-320
	PLOWP						
	LDPCHK		● 当特定文件名称的程序被执行时进入导通状态 ● 当特定文件名称的程序不被执行时进入不导通状态		* 2 + n	—	7-310
	ANDPCHK						
	ORPCHK						

* :n([程序名称字符数]/2) 表示一步(小数部分舍入)

2.5.18 其他指令

表 2.35 其他指令

类别	指令符号	符号	详细处理过程	执行条件	基本步数	子集	参见章节
WDT 复位	WDT		● 在顺序程序中复位看门狗定时器		1	—	7-324
	WDTP						
定时时钟	DUTY				4	—	7-326
时间检查	TIMCHK		● 如果输入条件测量到的 ON 时间连续超过设定时间，则将由(D)指定的软元件转为 ON。			—	7-328
在 1—字节单元内进行直接读/写操作	ZRRDB				3	—	7-329
	ZRRDBP						
	ZRWRB				3	—	7-331
	ZRWRBP						
	ADRSET				3	—	7-333
	ADRSETP						
来自键盘的数字键输入	KEY		● 将电(S)指定的输入单元的 8 个数据点的 ASCII 数据，转换成十六进制值到(D1)指定号码的软元件后，并存储。		5	—	7-334
索引寄存器的批量保存	ZPUSH		● 将索引寄存器 Z0 到 Z15 的注释存储到从(D)指定的软元件开始的区域。		2	—	7-338
	ZPUSHP						
索引寄存器的批量恢复	ZPOP		● 将存储在(D)指定的软元件开始的区域内的数据读到索引寄存器 Z0 到 Z15。				
	ZPOPP						
对 E2PROM 文件寄存器的批量写操作	EROMWR		● 将批数据写入到 E2PROM 文件寄存器内。		5	—	7-342
	EROMWRP						

2.5.19 用于数据链接的指令

表 2.36 用于数据链接的指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参见 章节
网络刷新	ZCOM	J.ZCOM Jn	刷新指定的网络		5	—	8-76
		JP.ZCOM Jn					
		G.ZCOM Un					
		GP.ZCOM Un					
QnA 链接指令：从另一个站读取数据	READ	J.READ Jn S1 S2 D1 D2	将其他站的字软元件数据读到本站中。		9	—	8-15
		G.READ Jn S1 S2 D1 D2					
		JP.READ Jn S1 S2 D1 D2					
		GP.READ Un S1 S2 D1 D2					
	SREAD	J.SREAD Jn S1 S2 D1 D2 D3			10	—	8-21
		G.SREAD Un S1 S2 D1 D2 D3					
QnA 链接指令：将数据写入到另一个站	WRITE	J.WRITE Jn S1 S2 D1 D2	将本站的数据写入到其他站的字软元件中。		10	—	8-27
		G.WRITE Un S1 S2 D1 D2					
		JP.WRITE Jn S1 S2 D1 D2					
		GP.WRITE Un S1 S2 D1 D2					
	SWRITE	J.SWRITE Jn S1 S2 D1 D2 D3			11	—	8-34
		G.SWRITE Un S1 S2 D1 D2 D3					
QnA 链接指令：发送数据	SEND	J.SEND Jn S1 S2 D1	发送数据(消息)到其他站		8	—	8-41
		G.SEND Un S1 S2 D1					
		JP.SEND Jn S1 S2 D1					
		GP.SEND Un S1 S2 D1					
QnA 链接指令：接收数据	RECV	J.RECV Jn S1 S2 D1	接收送往本站的数据(消息)		8	—	8-46/49
		G.RECV Un S1 S2 D1					
		JP.RECV Jn S1 S2 D1					
		GP.RECV Un S1 S2 D1					
QnA 链接指令：来自其他站的瞬时请求	REQ	J.REQ Jn S1 S2 D1 D2	将一个瞬时请求送往其他站并执行它。		8	—	8-55
		G.REQ Un S1 S2 D1 D2					
		JP.REQ Jn S1 S2 D1 D2					
		GP.REQ Un S1 S2 D1 D2					

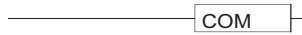
表 2.36 用于数据链接的指令(续表)

类别	指令符号	符号	详细处理过程	执行条件	基本步数	子集	参见章节
QnA 连接指令:从远程 I/O 站中的特殊功能模块中读取数据	ZNFR		从远程 I/O 站中的特殊功能模块中读取数据		8	—	8-67
QnA 连接指令:写数据到远程 I/O 站中的特殊功能模块	ZNT0		写数据到远程 I/O 站中的特殊功能模块		8	—	8-72
A-系列兼容连接指令:从其他站中读取软元件数据	ZNRD		将其他站中的软元件数据读取到本站。		32	—	8-77 8-81
A-系列兼容连接指令:写软元件数据到其他站。	ZNWR		将本站的数据写入到其他站的软元件中。		32	—	8-84 8-88
A-系列兼容连接指令:从远程 I/O 站的特殊功能模块中读取数据	RFRP		从远程 I/O 站的特殊功能模块中读取数据		11	—	8-91
A-系列兼容连接指令:写数据到远程 I/O 站的特殊功能模块中	RTOP		写数据到远程 I/O 站的特殊功能模块中		11	—	8-95
读取路由信息	RTREAD		读取路由参数设定数据		7	—	8-99
登记路由信息	RTWRITE		写路由数据到由路由参数指定的区域内		8	—	8-103

2.5.20 QCPU 指令





表 2.37CPU 指令

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
读取模块信息	UNIRD		● 读取存储在从(n)指定号码的 I/O 开始的区域内, 由(n2)指定点数的模块信息, 并存储它们到(d)指定的软元件开始的区域内。		4	—	9-2
	UNIRDP						
跟踪设定	TRACE		● 当 SM800, SM801, 和 SM802 变为 ON 时, 将设定在外部软元件上的跟踪数据存储到指定号码的内存卡中的跟踪文件内。		1	—	9-6
跟踪复位	TRACER		● 将由 TRACE 指令设置的数据复位。		1	—	9-6
写数据到指定的文件	SP.FWRITE		● 写数据到指定的文件		11	—	9-8
从指定的文件中读取数据	SP.FREAD		● 从指定的文件中读取数据		11	—	9-16
从存储卡中装载程序	PLOADP		● 将存储在存储卡或标准存储器(驱动器 0 外的其他软元件)中的程序传送到驱动器 0, 并将程序置于备用状态。		3	—	9-27
从程序存储器中卸载程序	PUNLOADP		● 删除存储在标准存储器(驱动器 0)中的备用程序。		3	—	9-30
装载+卸载	PSWAPP		● 删除存储在由(S1)指定的标准存储器(驱动器 0)中的备用程序。然后, 将存储在存储卡或由(S2)指定的标准存储器(驱动器 0 外的其他软元件)中的程序传送到驱动器 0, 并将程序置于备用状态。		4	—	9-33
文件寄存器中的高速块传送	RBMV		● 将由(S)指定的软元件的 n 数据点的 16 位数据传送到由(D)指定的软元件开始的区域内。		4	—	9-36
	RBMVp						
写入到本站 CPU 的共享存储器	S.TO		● 将本站的软元件数据写入到主机站 CPU 模块的共享存储区域内。		5	—	9-41
	SP.TO						
	TO		● 将本站的软元件数据写入到主机站 CPU 模块的 CPU 共享存储器内。		5	—	9-44
	TOP						
从其他站 CPU 的共享存储器中读取数据	FROM		● 将其他站 CPU 模块的 CPU 共享存储器区域内的软元件数据读取到主机站。		5	—	9-48
	FROMP						

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
CPU 共享存储器的自动刷新	COM		<ul style="list-style-type: none"> ● 执行智能功能模块的自动刷新，一般数据处理，和 CPU 共享存储器的自动刷新。 		1	—	9-50

2.5.21 冗余系统指令(用于 Q4ARCPU)

表 2.38 冗余系统指令(用于 Q4ARCPU)

类别	指令 符号	符号	详细处理过程	执行条件	基本 步数	子 集	参 见 章 节
在 CPU 启动时的操作模式设置	S. STMODE		<ul style="list-style-type: none"> ● 当系统上电进行 CPU 启动时，确定 (S1) 上的操作模式，在启动前是清除 Q4ARCPU 还是不清除它们。 		9	—	10-2
CPU 切换过程中的操作模式设置指令	S. CGMODE		<ul style="list-style-type: none"> ● 当控制由主控系统切换到备用系统时，确定 (S1) 上的操作模式，在启动前是清除 Q4ARCPU 软元件还是不清除它们。 		7	—	10-4
数据跟踪	S. TRUCK		<ul style="list-style-type: none"> ● 在 END 处理过程中，按照存储在由 (S) 指定的软元件开始的区域内的参数块数据内容，对软元件存储器进行跟踪。 		6	—	10-6
缓冲存储器的批量刷新	S. SPREF		<ul style="list-style-type: none"> ● 按照存储在由 (S) 指定的软元件开始的区域内的参数块数据内容，对特殊功能模块缓冲存储器内的内容进行批量读/写。 		6	—	10-10

3. 指令结构

3.1 指令结构

多数的 CPU 模块指令包括一个指令部分和一个软元件部分。

- 指令部分..... 显示指令的功能。
- 软元件部分.... 显示指令所需的数据。

软元件部分被分为源数据，目标数据和软元件号。

(1) 源数据⑤

(a) 源数据是运算所需的数据。

(b) 根据指定的软元件不同，源数据具有以下形式：

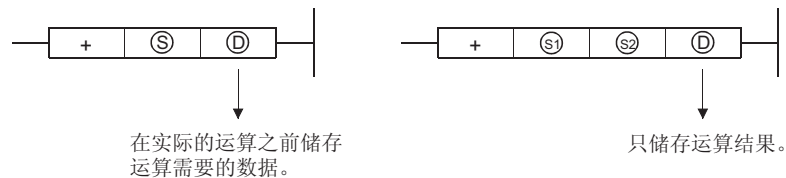
- 常数..... 指定将在运算中使用的数值。当程序写好以后设定，在程序执行过程中不能改变。当常数需要被当作变量使用时，需要索引。
- 位软元件和字软元件..... 指定存储运算用数据的软元件。数据必须存储在指定的软元件当中，直到运算开始执行。通过在程序执行过程中改变指定软元件中存储的数据，在指令中要使用的数据就可以更改。

(2) 目标数据⑥

(a) 目标数据存储运算执行后的数据。

然而，一些指令要求在运算执行之前，在目标数据中存储即将在运算中使用的数据。

例如：一个包含有 BIN16-位数据的加指令。



(b) 供数据存储用的软元件必须被设置到目标中去。

- (3) 软元件号和传输号(n)
- (a) 软元件号和传输号指定了在多软元件指令中要用到的软元件号和传输号。
- 例如：块传输指令

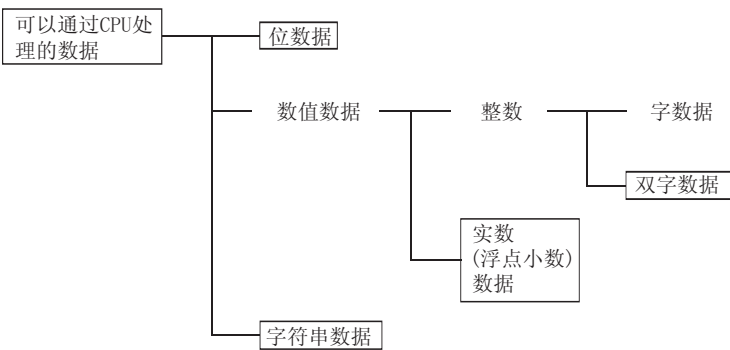


- (b) 软元件号或传输号可以设定在 0 和 32767 之间，然而，如果该号为 0，该指令将为无运算指令。

3

3.2 数据的指定方法

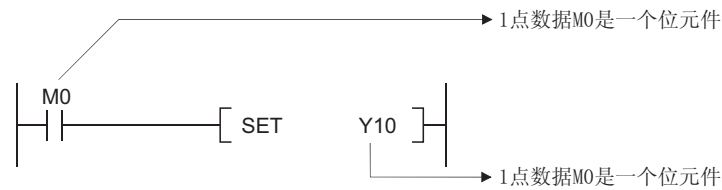
以下五种数据可以用于 CPU 模块指令：



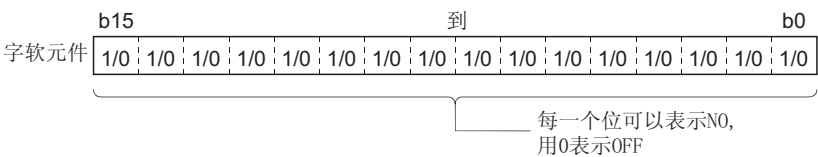
3.2.1 使用位数据

位数据是指以一位为单位使用的数据，如触点或线圈。“位软元件”和“位指定字软元件”可以被当作位数据使用。

- (1) 使用位软元件时
- 位软元件在一点数据单元中指定。



- (2) 使用字软元件时
- (a) 字软元件通过指定位号，使得指定位号的 1/0 被用于位数据。

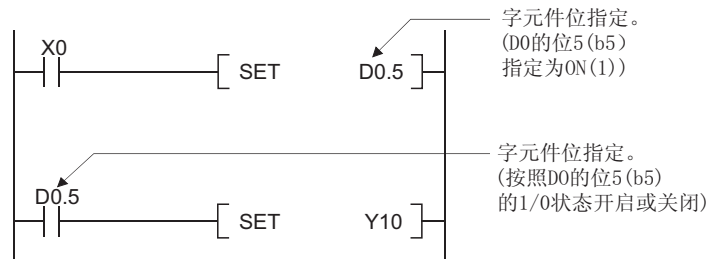


(b) 字软元件的位指定是通过指定 “ 字软元件 位号 ” 来完成的。

(位号指定在十六进制中完成)。

例如：D0 的位 5 (b5) 指定为 D0.5，D0 的位 10 (b10) 指定为 D0.A。

然而，对于定时器(T)，累计定时器(ST)，计数器(C)或索引寄存器(Z)而言，可能就没有位指定(如 Z0.0 就不存在)。



3.2.2 使用字(16 位)数据

字数据是基本指令和应用指令使用的 16 位数据

以下两种形式的字数据可以在 CPU 模块中使用：

- 十进制常数..... K-32768 (到) K32767
- 十六进制常数..... H0000 (到) HFFFF

由数字指定的字软元件和位软元件可以作为字数据使用。

对于直接的访问输入 (DX) 和直接的访问输出 (DY)，字数据不能通过数字输出 (DR) 指定。(关于直接访问输入和直接访问输出，请参考使用的 CPU 模块的用户手册(功能解释，编程基础篇)或 QnACPU 编程手册(基础篇))

(1) 当使用位软元件时

(a) 当数位被指定后，位软元件就可以处理字数据。

位软元件的数位指定是通过指定 位号 和 位软元件的初始号 来完成的。位软元件的数位指定可以在 4 点(4 位)单元中完成，并且可以指定为 K1 到 K4。(对于链接直接软元件，指定是通过 “J 网络号 \ 数位指定 / 位软元件的初始号 ” 完成的。如 X100 到 X10F 用于指定网络号 No. 2，它是通过 J2\K4X100 来完成的)。

例如，如果 X0 被用于数位指定，下列各点需要指定：

- K1X0 从 X0 到 X3 的 4 点被指定
- K2X0 从 X0 到 X7 的 8 点被指定
- K3X0 从 X0 到 XB 的 12 点被指定
- K4X0 从 X0 到 XF 的 16 点被指定

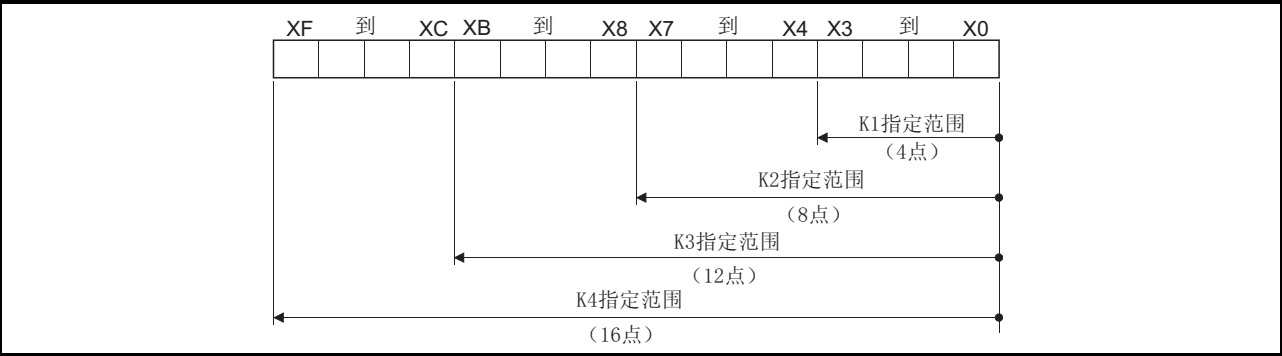


图 3.1 16 位指令的数位指定设定范围

(b) 如果在源数据⑤中数位指定已经完成，在表 3.1 中的所显示的数值就可以当作源数据进行处理。

表 3.1 当作数位指定处理的数值表

指定的数位号	通过 16 位指令
K1 (4 点)	0 到 15
K2 (8 点)	0 到 255
K3 (12 点)	0 到 4095
K4 (16 点)	-32768 到 2767

如果在源位置是由数位指定来指定的位软元件，并且目标是字软元件，那么从由源数据中的数位指定来指定的位之后，目标数据中的字软元件为 0。

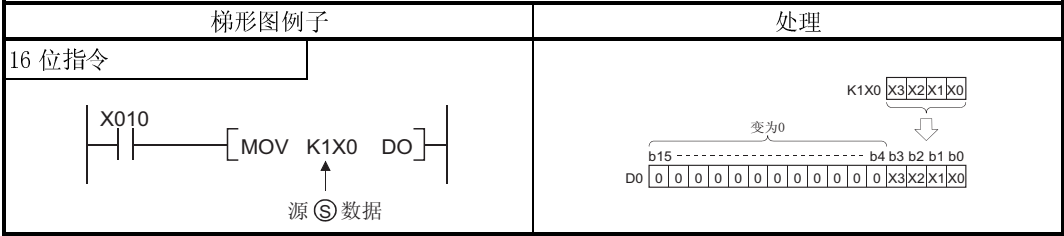


图 3.2 梯形图例子和执行的处理

(c) 一旦在目标⑥中完成数位指定，那么指定的点数作为目标使用。在指定为数位的点数之后的位软元件不变。

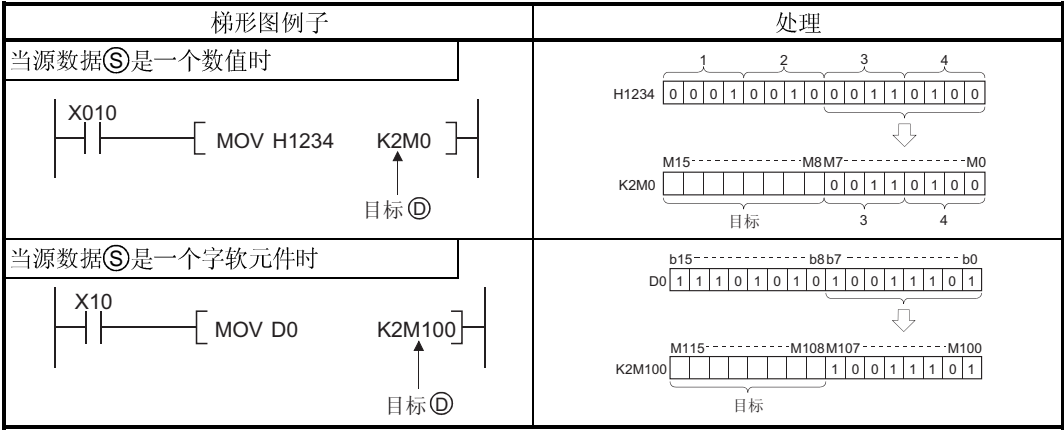
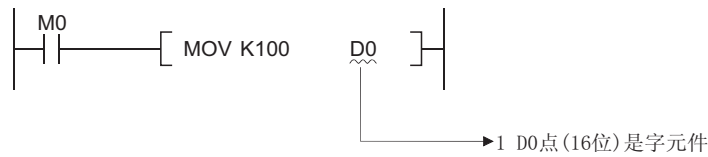


图 3.3 梯形图例子和进行的处理

- (2) 当使用字软元件时
字软元件在 1 点单位 (16 位) 中被指定。



要点
(1) 进行数位指定处理时，位软元件初始软元件号可以为任意数值。 (2) 数位指定不可用于直接访问 I/O (DX, DY)。

3.2.3 使用双字数据 (32 位)

双字数据是被基本指令和应用指令使用的 32 位数值数据。两种类型的双字数据可以被 CPU 模块处理，如下所示：

- 十进制常数..... K-2147483648 到 K2147483647
- 十六进制常数..... H00000000 到 HFFFFFFF

由数位指定来指定的字软元件和位软元件可以当作双字数据使用。
对于直接访问输入 (DX) 和直接访问输出 (DY)，双字数据的指定是不可能通过数位指定来完成的。

- (1) 使用位软元件时
- (a) 数位指定可以使位软元件处理双字数据。
- 位软元件的数位指定是通过指定 数位号 和 初始位软元件号 来完成的。位软元件的数位指定可以在 4 点 (4 位) 单元中完成，可用于 K1 到 K8 的指定。(对于链接直接软元件，指定是通过 “J 网络号 \ 数位指定初始位软元件号” 来完成的。当 X100 到 X11F 为网络号 2 指定时，是通过 J2\K8X100 来完成的)。
- 例如，如果 X0 被指定用于数位指定，下列各点就可以被指定：
- K1X0 X0 到 X3 的 4 点被指定
 - K2X0 X0 到 X7 的 8 点被指定
 - K3X0 X0 到 XB 的 12 点被指定
 - K4X0 X0 到 XF 的 16 点被指定
 - K5X0 X0 到 X13 的 20 点被指定
 - K6X0 X0 到 X17 的 24 点被指定
 - K7X0 X0 到 X1B 的 28 点被指定
 - K8X0 X0 到 X1F 的 32 点被指定

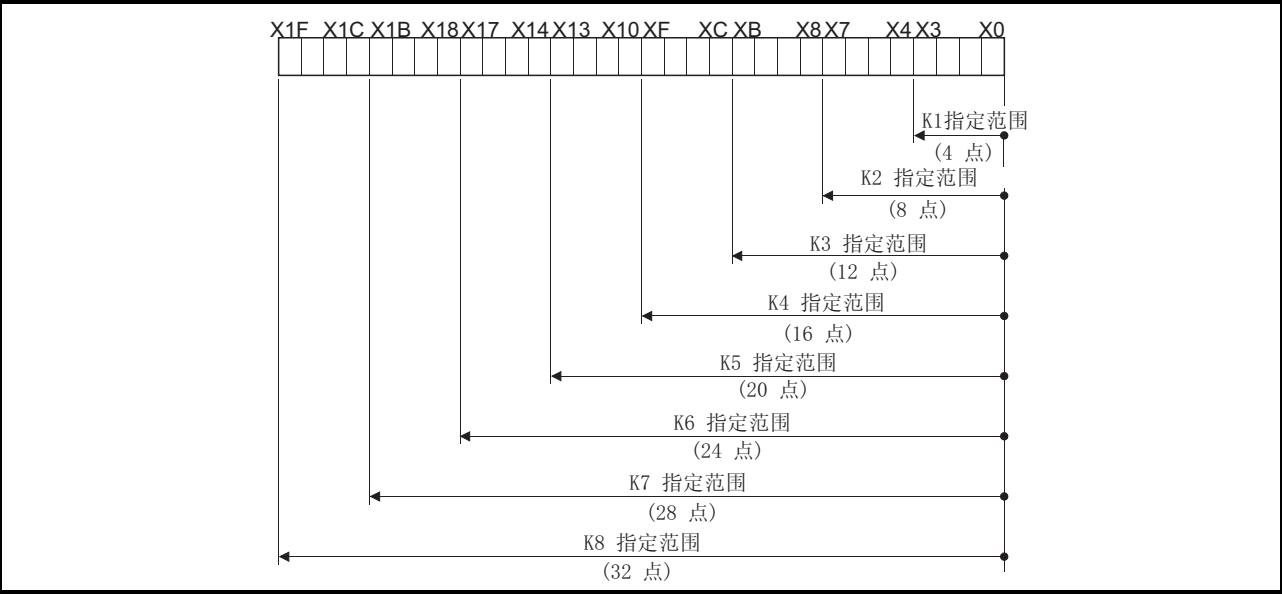


图 3.4 32 位指令位指定设定范围

(b) 如果在源数据⑤位置数位指定已经完成，在表 3.2 中所显示的数值就可以当作源数据来处理。

3.2 可以作为数位指定处理的数值列表

指定的数位号	对 32 位指令	指定的数位号	对 32 位指令
K1(4 点)	0 到 15	K5(20 点)	0 到 1048575
K2(8 点)	0 到 255	K6(24 点)	0 到 16777215
K3(12 点)	0 到 4095	K7(28 点)	0 到 268435455
K4(16 点)	0 到 65535	K8(32 点)	-2147483648 到 2147483647

如果在源位置是由数位指定的一位软元件，并且目标是字软元件，那么在目标数据中，由源数据中数位指定所指定的位以后，目标数据的字软元件变为 0。

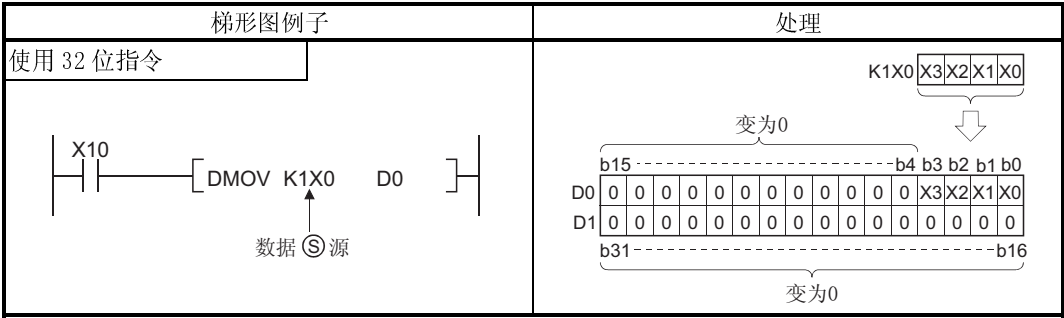


图 3.5 梯形图例子和进行的处理

(c) 一旦在目标⑤的位置完成了数位指定，指定的点数可以当作目标来使用。
 点数指定为数位以后，位软元件不能变更。

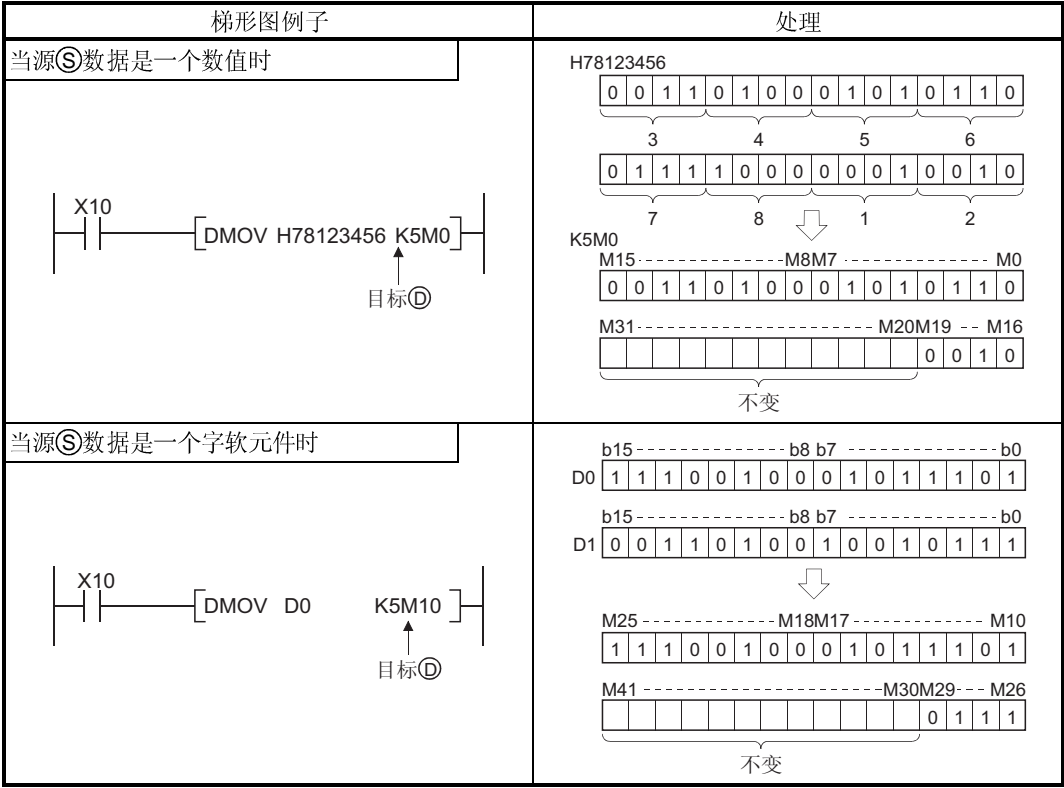
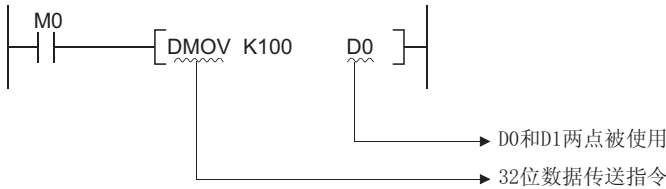


图 3.6 梯形图例子和进行的处理

要点
(1) 进行位指定处理时，位软元件初始软元件号是一个随机的值。
(2) 数位指定不可用于直接访问 I/O (DX, DY)。

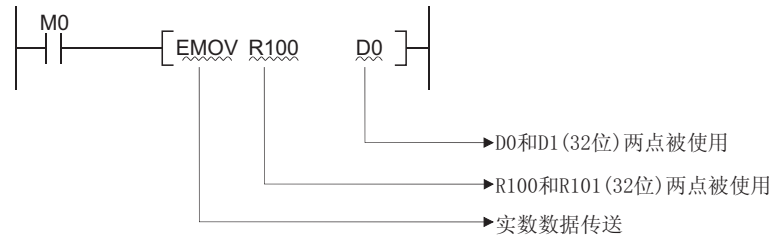
- (2) 当使用字软元件时
- 字软元件可指定被数据的低 16 位使用的软元件。
- 一个 32 位指令使用 (指定软元件号) 和 (指定软元件号+1)。



3.2.4 使用实数数据

实数数据是用于基本指令和应用指令的 32 位浮点小数数据。
只有字软元件能够存储实数数据。

处理实数数据的指令指定那些被更低的 16 位数据使用的软元件。实数存储在 32 位中，可以弥补 (指定的软元件号) 和 (指定的软元件号+1)。

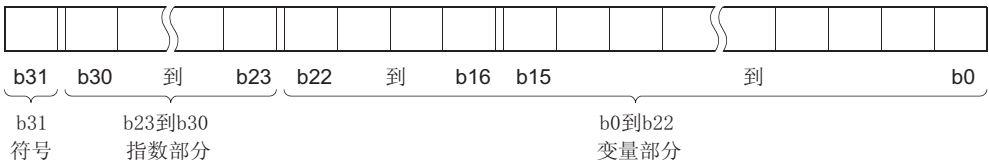


备注

1) 在顺控程序中，实数通过 E 指定

浮点小数数据使用两个字软元件并以下列方式表达：
[符号]1.[变量部分]×2[指数部分]

位配置和浮点小数数据的内部代表意义如下：



- 符号
符号在 b31 处被表示
0: 正
1: 负
- 指数部分
2ⁿ 中的 n 是由 b23 到 b30 表示
根据 b23 到 b30 的 BIN 值的不同，n 的值如下所示：

b23到b30	FFH	FEH	FDH		81H	80H	7FH	7EH		02H	01H	00H
n	非数字	127	126		2	1	0	-1		-125	-126	非数字

- 变量部分
从 b0 到 b22 的 23 位，在二进制位置 1. XXXXXX. 代表. XXXXXX...

要点
<p>(1) 通过外围设备的监视功能，CPU 模块的浮点小数数据能够被监视。</p> <p>(2) 当浮点数据被用于表达 0 时，从 b0 到 b31 所有的数据均变为 0。</p> <p>(3) 所有的实数设定范围是 0 和 $\pm 2^{-126} \leq \text{值} < 2^{128}$。</p> <p>(4) 不要在浮点数据中指定-0（当只有浮点实数的最有效位为 1 时）（如果浮点运算是通过-0 进行的，那么将会出现一个运算错误）。</p> <p>CPU 模块进行双精度浮点运算的内部运算时不会出现运算错误，这是因为在-0 被指定时，在 CPU 模块中将-0 转换为 0，然后再执行的浮点运算。进行单精度浮点运算的内部运算的 CPU 模块会出现运算错误，因为它赋予处理速度优先权，并且在浮点运算时使用-0，在-0 被指定时没有转换。</p> <p>(a) 当-0 被指定时，下列 CPU 模块将不会导致运算错误。</p> <p>内部运算设定为双精度的高性能模型 QCPU *1 (浮点运算的内部运算默认为双精度)</p> <ul style="list-style-type: none">● QnACPU● Q2ACPU● Q4ARCPU (它的 SM707 处于开启状态) <p>(b) 当-0 被指定时，下列 CPU 模块将导致运算错误</p> <ul style="list-style-type: none">● 基本模型 QCPU*2● 内部运算设置成单精度的高性能模型 QCPU *1● 过程 CPU● Q4ARCPU (它的 SM707 I 处于关闭状态)

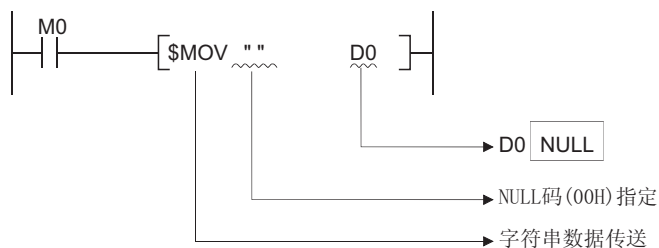
备注

- * 1: 在 PLC 参数对话框中的 PLC 系统设定中，在浮点运算的内部运算的单精度和双精度之间进行转换。
- 对于浮点运算的单精度和双精度，请参见高性能模型 QCPU(Q 模式)的用户手册(功能解释，编程基础)。
- * 2: 如果序列号的头五位为“04122 或更大”的基本模型 QCPU 能够执行浮点运算。

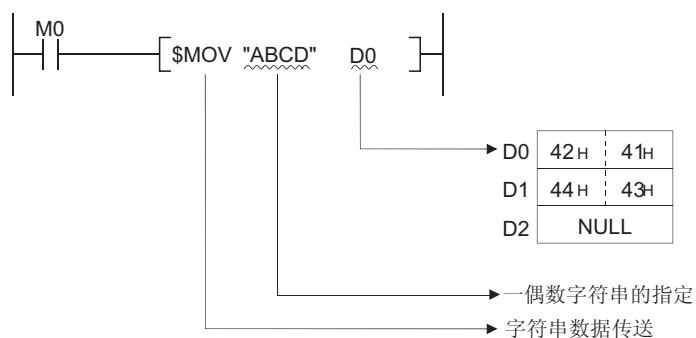
3.2.5 使用字符串数据

字符串数据是基本指令和应用指令使用的字符数据。
它包含从指定字符到 NULL 码 (00H) 的所有数据。

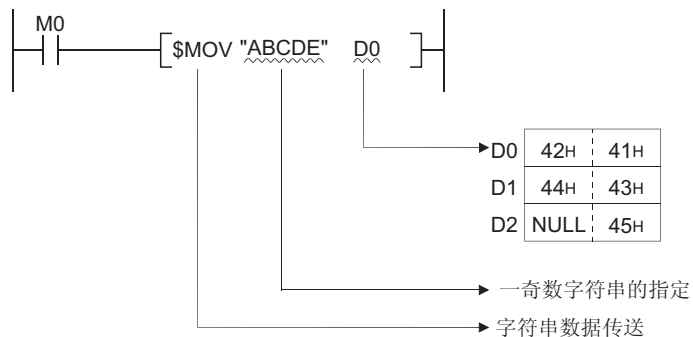
- (1) 当指定字符是 NULL 码时。
使用一个字来存储 NULL 码。



- (2) 当字符数是偶数时
使用 (字符数/2+1) 字，并且存储字符串和 NULL 码。
例如，如果 “ABCD” 被转换成 D0，那么 ABCD 就存储在 D0 和 D1 处，NULL 码存储在 D2 处。

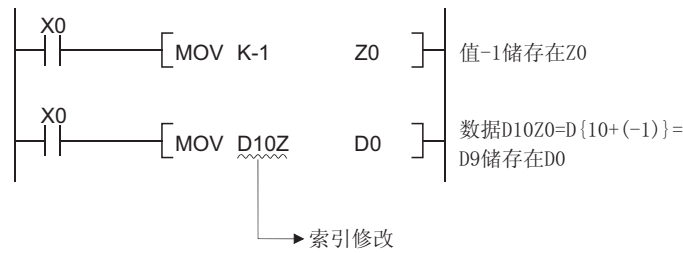


- (3) 当字符数是奇数时
使用 (字符数/2) 字 (小数部分四舍五入)，并且存储字符串和 NULL 码。
例如，如果 “ABCDE” 传送到 “D0”，字符串 (ABCDE) 和 NULL 码将会从 D0 存储到 D2 上去。



3.3 索引修改

- (1) 索引修改
- (a) 索引修改是通过使用索引寄存器来完成的间接设定。
- 当在一个顺控程序中使用索引修改时，将要使用的软元件将会成为指定的软元件号直接加上索引寄存器的内容。例如，如果 D2Z2 已经被指定，那么指定的软元件计算如下：
- $D(2+3)=D5$ 和 Z2 的内容为 3，成为指定的软元件。
- (b) 这里有 16 个索引寄存器，从 Z0 到 Z15。
- 每一个索引寄存器都能在-32768 到 32767 之间设定。



示例

在已经执行索引修改的场合，实际处理软元件如下所示：（当 Z0=20 并且 Z1=-5）

梯形图例子	实际处理软元件

图 3.7 梯形图例子和实际处理软元件

- (2) 可以进行索引修改的软元件
- 除去下面所列的限制，索引修改可以应用于触点，线圈，基本指令和应用指令中使用的软元件。
- (a) 不能使用索引修改的软元件

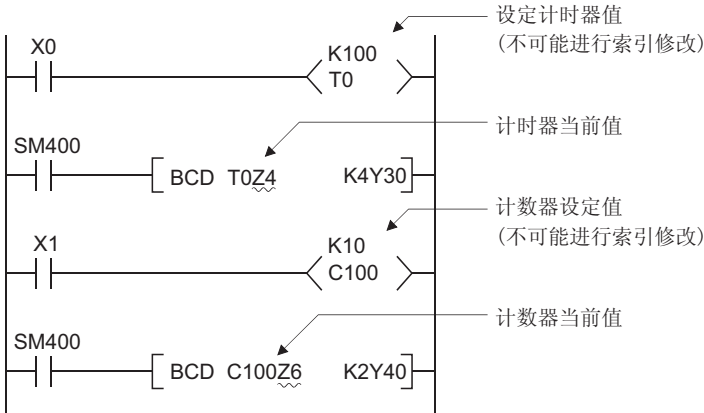
软元件	含义
K, H	32 位常数
E	浮点小数数据
\$	字符串数据
[]	为字软元件指定的位
FX, FY, FD	功能软元件
P	用做标记的指针
I	用做标记的中断点
Z	索引寄存器
S	步进继电器
TR	SFC 传送软元件 *1
BL	SFC 块软元件 *1
T, ST	计时器设定值
C	计数器设定值

(b) 和索引寄存器一起使用的软元件的限制

软元件	含义	应用例
T	● 只有 Z0 和 Z1 可以用于计时器触点和线圈	
C	● 只有 Z0 和 Z1 可以用于计数器触点和线圈	

备注

- 1) * 1: SFC 传送软元件和 SFC 块软元件是供 SFC 使用的软元件。
关于如何使用这些软元件，请参考 QCPU (Q 模式)/QnACPU 编程手册 (SFC)
- 2) 对于计时器和计数器的当前值，没有索引寄存器使用数量的限制。

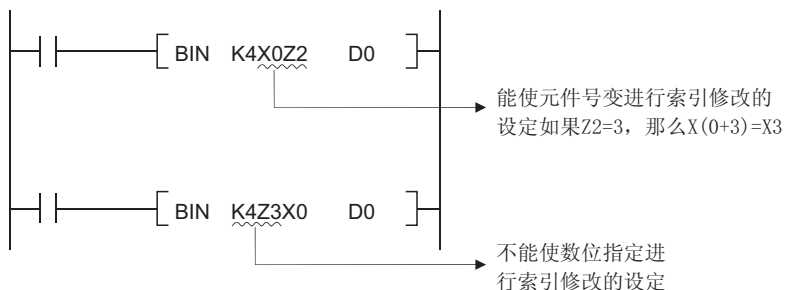


(c) 其他

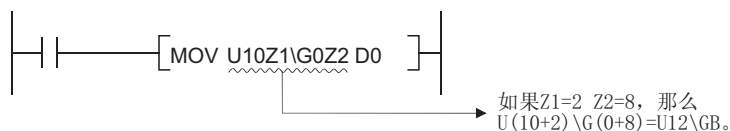
1) 位数据

在执行数位指定时，软元件号可以进行索引修改。

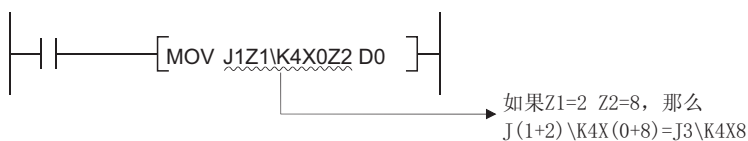
然而，索引修改不可能由数位指定来完成的。



2) I/O 地址号和缓冲存储器可以用特殊功能模块软元件进行索引修改。



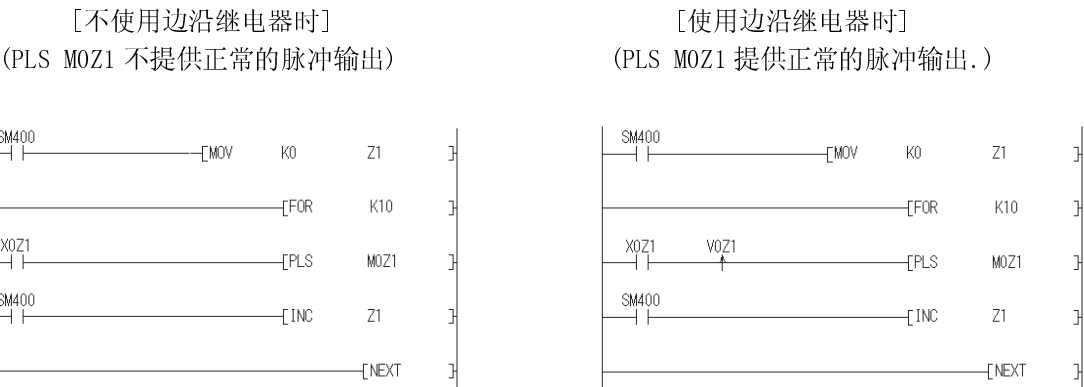
3) 网络号和软元件号可以使用链接直接软元件来进行索引修改



备注

- 1) *1: 关于特殊功能模块软元件，请参考使用的 CPU 模块的用户手册(功能解释，编程基础篇)或 QnACPU 编程手册(基础篇)。
- 2) *2: 对于直接链接软元件，请参考使用的 CPU 模块的用户手册(功能解释，编程基础篇)或 QnACPU 编程手册(基础篇)。

- (1) 在 FOR 和 NEXT 指令之间进行索引修改的注意事项。
- 当对 PLS/PLF，或只在 FOR 和 NEXT 指令之间 OFF 到 ON (□P) 的前沿上执行的指令的软元件进行索引时，使用边沿继电器 (V) 作为执行条件。
- 如果 PLS/PLF/只在 FOR 和 NEXT 指令之间 OFF 到 ON (□P) 的前沿上执行的指令的软元件在没有使用边沿继电器的情况下进行索引，那么 PLS/PLF/只在 OFF 到 ON (□P) 的前沿上执行的指令不会正常动作。



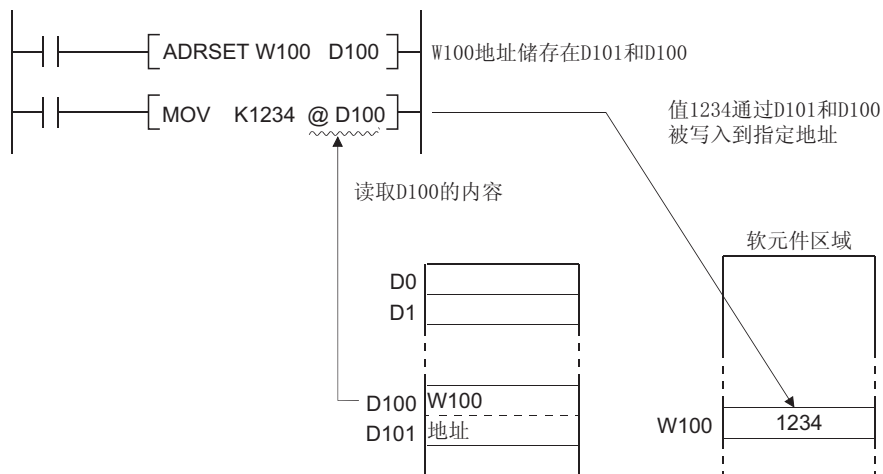
备注

- * 1: ON/OFF 数据由边沿继电器 V0Z1 存储。
- 例如，X0 的 ON/OFF 数据由 V0 存储，X1 的数据由 V1 存储。

3.4 间接指定

(1) 间接指定

- (a) 间接指定是一种通过使用字软元件来指定软元件地址的方法，这种方法会在顺控程序中使用。在索引寄存器不足的情况下，可以使用这种方法。
- (b) 用于指定指定软元件地址的软元件是通过“@ + (字软元件号)”来指定的。
例如，@D100 的指定将会使得 D100 D101 的内容成为软元件地址。
- (c) 执行间接指定的软元件地址可以通过 ADRSET 指令来进行确认。



- (2) 能间接指定的软元件。
- 能间接指定的 CPU 模块软元件在表 3.3 中列出：

表 3.3 能间接指定的软元件列表

软元件类型		能否被间接指定	间接指定范例
内部用户软元件	位软元件 *1	否	—
	字软元件*1	能	● @ D100 ● @ D100Z2 *2
链接直接软元件	位软元件*1	否	—
	字软元件*1	能*3	● @ J1\W10 ● @ J1Z1\W10Z2 *2
特殊直接软元件		能*3	—
索引寄存器		否	● @ R0, @ ZR20000 ● @ R0Z1, @ ZR20000Z1 *2
文件寄存器		能	—
嵌套结构		否	—
指针			—
常数			—
其他	SFC 块软元件		—
	SFC 之后的软元件		
	网络号		
	I/O 号		

备注

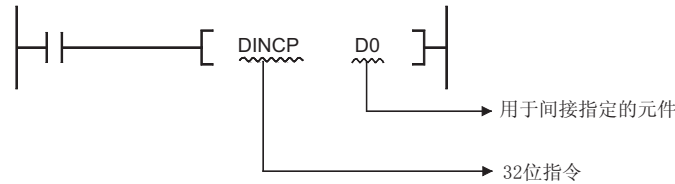
- 1) *1：关于软元件名称，请参考使用的 CPU 模块用户手册(功能解释，编程基础篇)或 QnACPU 编程手册(基础篇)
- 2) *2：表示通过索引寄存器进行间接索引修改。
- 3) *3：软元件能够间接指定，然而地址不能写入 ASRSET 指令中。
- (3) 注意事项

间接指定的地址需要使用两个字来指定。

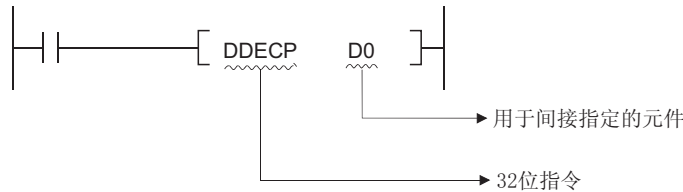
因此，用间接指定来替换索引修改，要求使用 32 位数据的加減

以下是在间接指定中，为存储在 D1 和 D0 中的软元件地址的加減而使用的梯形图。

[在间接指定中，增加“1”到软元件的地址中]



[在间接指定中，从软元件地址中减少“1”]



3.5 子集处理

子集处理是用来对基本指令和应用指令中使用的位软元件设定限制的，目的是为了加快处理速度。但是，指令符号不变。
为了缩短扫描周期，请在下面所列出的条件下运行指令。

(1) 每一软元件为子集处理必须满足的条件

(a) 当使用字数据时

软元件	条件
位软元件	<ul style="list-style-type: none"> ● 以 16 的因数来指定一个位软元件数 ● 只有 K4 可以在数位指定中被指定 ● 不能进行索引修改
字软元件	● 内部软元件(不包括文件寄存器 ZR)
常数	● 没有限制

(b) 当使用双字数据

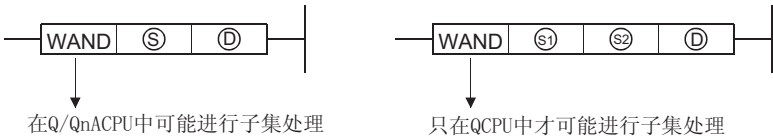
软元件	条件
位软元件	<ul style="list-style-type: none"> ● 以 32 的因数来指定一个位软元件数 ● 只有 K8 可以在数位指定中被指定 ● 不能进行索引修改
字软元件	● 内部软元件(不包括文件寄存器 ZR)
常数	● 没有限制

(2) 子集处理中能够用到的指令

指令类型	指令符号
比较指令	● =, <, >, <=, >=, D=, D<, D>, D<=, D>=, D>=
基本的数学运算(加减乘除)	<ul style="list-style-type: none"> ● +, -, *, /, INC, DEC, D+, D-, D*, D/, DINC, DDEC ● B+, B-, B*, B/
数据变换指令	● BCD, BIN, DBCD, DBIN
数据传送指令	<ul style="list-style-type: none"> ● MOV, DMOV, CML, DCML, XCH, DXCH ● FMOV, BMOV, EMOV (仅用于 QCPU)
程序分支指令	● CJ, SCJ, JMP
逻辑运算*	● WAND, DAND, WOR, DOR, WXOR, DXOR, WXNR, DXNR
循环指令	● RCL, DRCL, RCR, DRCR, ROL, DROL, ROR, DROR
移动指令	● SFL, DSFL, SFR, DSFR
数据处理指令	● SUM, SEG
结构化程序指令	● FOR, CALL

备注

1) *：只有 QCPU 才能使用三个软元件来进行逻辑运算指令 WAND, DAND, WOR, DOR, WXOR, DXOR, WXNR, 或 DXNR 的子集处理



3.6 编程注意事项(操作错误)

当在 CPU 模块中执行基本指令和应用指令时，在下列例子中将会被返回操作错误：

- 每个指令的出错信息将会被列在解释页上。
- 当使用缓存寄存器，智能模块或特殊功能模块没有安装在指定的 I/O 号位置。
- 当使用一个链接软元件时，不存在相关的网络。
- 当使用一个链接软元件时，在指定的 I/O 号位置上没有安装网络模块。

要点
(1) 当文件寄存器设定已经完成但是没有安装内存卡时，或没有进行文件寄存器设定，即使试图将其写入文件寄存器中，出错也不会返回。 然而，如果尝试在这个文件寄存器(已经被尝试过写入操作)中读取“FFFFh”将会被存储。

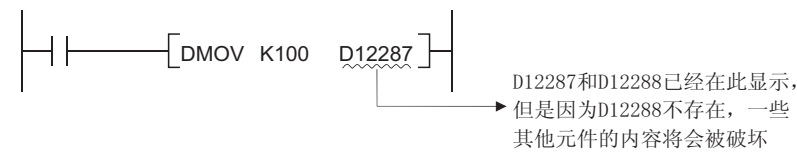
(1) 软元件范围检查

在 CPU 模块中，基本指令和应用指令使用的软元件范围检查，显示如下：

(a) 对于处理固定长度软元件(MOV, DMOV, 等)的指令不需要进行软元件范围检查。

如果相应的软元件范围被超出，数据会被写入其他软元件。*

例如，如果数据寄存器被分配了 12k 点，即使它超过了 D12287 点，也不会出现错误。

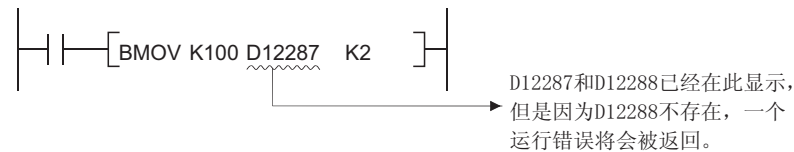


如果执行了索引修改，软元件范围检查也不会进行。

(b) 对于处理可变长度软元件(BMOV, FMOV 和其他指定传输号的软元件)的指令将会进行软元件范围检查。

一旦相应的软元件范围被超出，运算错误会返回。

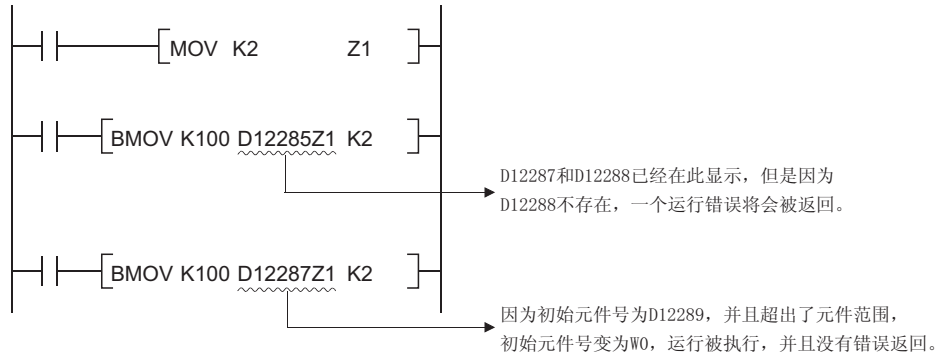
例如，当数据寄存器被分配了 12k 点，如果超过 D12287，将会出现错误。



备注

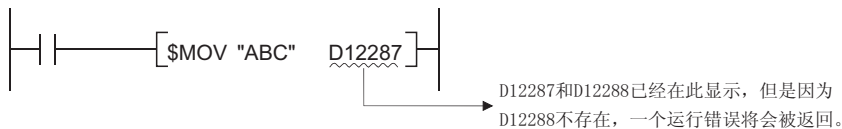
1) *：如果需要内部用户软元件分配顺序，请参见章节 3.4(3)

当执行了索引修改时，软元件范围检查同样也会进行。但是，如果已经执行了索引修改，并且初始软元件号超过了相关的软元件范围时，错误不会返回。



(c) 因为所有的字符串数据长度可变，软元件的范围检查都会进行，一旦相应的软元件字符串被超出了，运算错误会返回。

例如，数据寄存器被分配了 12 k 点，如果它超过了 D12287，将会出现错误。



请注意，即使由于索引修改的结果，起始软元件号超出了软元件范围，也不会出现任何运算错误。

(d) 如果通过直接的访问输出 (DY) 执行了索引修改，软元件范围检查也会进行。

(2) 软元件数据检查

对于 CPU 模块中基本指令和应用指令所要用的软元件，进行的软元件数据检查如下：

(a) 当使用 BIN 数据时

- 即使运算结果上溢或下溢，也不会有错误返回。
进位标志也同样不会继续多次进行。

(b) 当使用 BCD 数据时

- 1) BCD 值中 (0 到 9) 每一位均检查。如果某个数位超出了 0 到 9 (A 到 F) 的范围，一个运算错误将会返回。
- 2) 即使运算结果上溢或下溢，错误也不会返回。进位标志同样也不会继续多次进行。

(c) 当使用浮点十进制小数数据时

在下列情况下，运算错误会返回：

- 当浮点十进制小数数据值为 0
- 当浮点十进制小数数据的绝对值是 1.0×2^{-127} 或更低。
- 当浮点十进制小数数据绝对值是 1.0×2^{128} 或更高

(d) 当使用字符串数据时不进行任何数据检测。

- (1) 如果内部用户软元件分配被参数软元件分配变更，这种分配是在如下所示的软元件顺序中完成的：
- 如果这使用的种软元件分配少于 28.75 k 字，那么软元件对应的区域将为空。

初始地址 (固定)	SM
	SD
	X
	Y
	M
	L
	B
	F
	SB
	V
	S
	T 触点及线圈
	ST 触点及线圈
	C 触点及线圈
	T 的当前值
	ST 的当前值
	C 的当前值
	D
	W
	SW
	空区
	文件寄存器 (32K点)
当使用的元件少于28.75字时， 创建空区。	

备注

- 1) 如果需要更改内部用户软元件分配，请参考使用的 CPU 模块的用户手册(功能解释，编程基础篇)或 QnACPU 编程手册(基础篇)。

3.7 指令执行条件

- 对于 CPU 模块顺控指令，基本指令，和应用指令，存在以下四种类型的执行条件。
- 无条件执行..... 执行的指令与软元件 ON/OFF 状态无关。
例如：LD X0, OUT Y10
 - 在 ON 的状态下执行 在输入条件是 ON 的情况下执行指令。
例如：MOV 指令, FROM 指令
 - 在前沿执行..... 指令只在输入条件的前沿执行(当它从 OFF 转换到 ON 时)。
例如：PLS 指令, MOVP 指令
 - 在后沿执行..... 指令只在输入条件的后沿执行（当它从 ON 转换到 OFF 时）
例如：PLF 指令

对于线圈或与之同等的基本指令或应用指令，对于同样的指令既可以在 ON 状态下指定，也可以在前沿执行条件下指定，在指令名称后加上一个“P”用以说明执行条件。

- 在 ON 状态下执行的指令

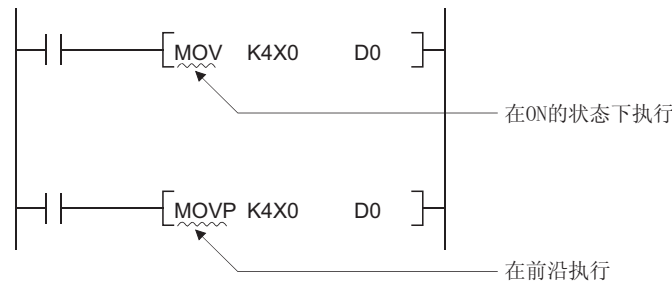
指令名

- 在前沿执行的指令

指令名

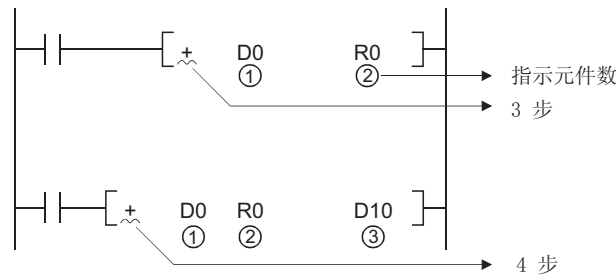
 + P

对 ON 状态下执行和在前沿条件下执行的 MOV 指令作如下指定：



3.8 计算步数

根据是否能间接设定使用的软元件，CPU 模块顺控指令，基本指令和应用指令的步数均有不同。
基本指令和应用指令的基本步数是通过将软元件号加 1 来计算出来的。
例如，“+指令”应该按照以下方法计算：



- (1) 增加步数的条件
- 当间接指定的软元件被使用时，或步数在增加时，步数将增加，超过基本步数。
- (a) 当软元件被间接指定时
- 当间接指定是通过 @ [] 来完成的，步数在基本步数上增加一步。
- 例如：当一个 3 步的 MOV 指令被间接指定时（例如 MOV K4X0 @D0），将会增加一步从而指令变为四步。

(b) 步数增加的软元件

步数增加的软元件	增加的步数	示例
智能模块软元件/特殊功能模块软元件	1	MOV U4\G10 D0
链接直接软元件		MOV J3\B20 D0
序列号访问文件寄存器		MOV ZR123 D0
32 位常数		DMOV K123 D0
实数常数		EMOV E0.1 D0
字符串常数	偶数情况： (字符数)/2 奇数情况： (字符数 + 1)/2	\$MOV "123" D0

- (c) 如果在(a)和(b)中描述的条件重叠，那么步数将增加二。
- 例如，如果 MOV U1\G10 ZR123 被指定，缓存寄存器指定将会被增加一步，序列号访问文件寄存器指定也将会被增加一步，从而在总步号上增加两步。

3.9 当 OUT, SET/RST, 或 PLS/PLF 指令使用相同的软元件时的操作

以下描述如何在一次扫描周期中执行 OUT, SET/RST 或使用相同软元件的 PLS/PLF 的多个指令。

(1) 使用相同软元件的 OUT 指令：

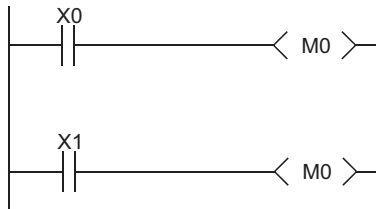
在编程过程中，一个扫描周期使用相同的软元件的 OUT 指令不要超过一次。

如果在一个扫描周期中有使用相同软元件的 OUT 指令，那么根据相关 OUT 指令的程序操作结果，每次 OUT 指令执行时，指定的软元件都将开启或关闭。

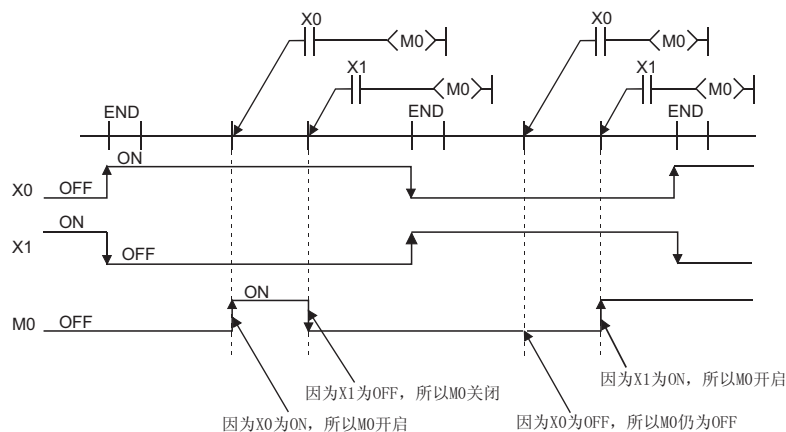
由于每个软元件的 ON 或 OFF 取决于每个 OUT 指令的执行结果，这样在一个扫描周期内软元件可能重复变为 ON 和 OFF。

以下图表为电路示例，显示如何开启/关闭带有输入 X0 和 X1 的同一个内部继电器 (M0)。

[电路]



[时序图]



对于刷新型 CPU 模块，当输出 (Y) 被 OUT 指令指定后，扫描周期的最后一个 OUT 指令的 ON/OFF 状态将被输出。

(2) 使用相同软元件的 SET/RST 指令

(a) 当 SET 指令为 ON 时，SET 指令开启指定的软元件。当 SET 指令为 OFF 时，SET 指令不进行任何操作。

为此，当两个或更多的 SET 指令在一个扫描周期中使用相同的软元件时，如果其中任何一个 SET 指令为 ON，那么指定的软元件将会为 ON。

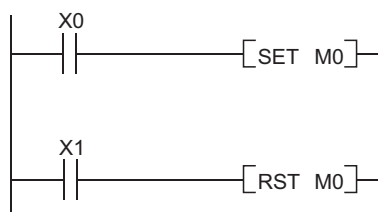
(b) 当 RST 指令为 ON 时，RST 指令关闭指定的软元件。当 RST 指令为 OFF 时，RST 指令不进行任何操作。

为此，当两个或更多的 RST 指令在一个扫描周期中使用相同的软元件时，如果任何一个 RST 指令为 ON，那么指定的软元件将会为 OFF。

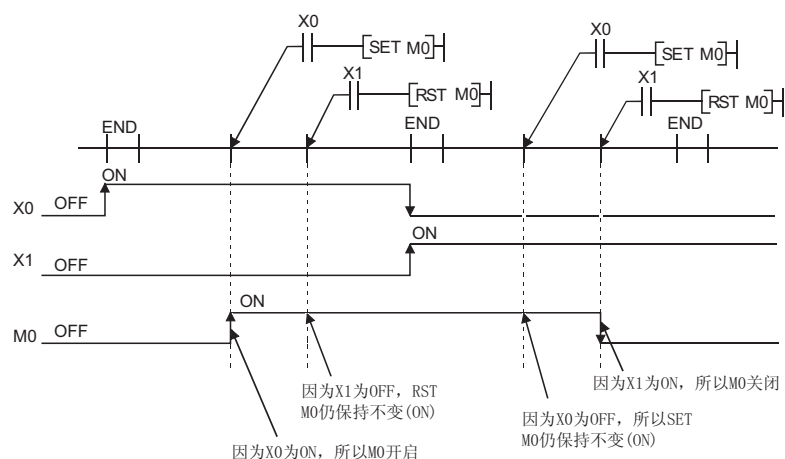
(c) 如果使用同一个软元件的 SET 指令和 RST 指令编在一个扫描周期中，那么当 SET 指令为 ON 时，SET 指令开启指定的软元件。当 RST 指令为 ON 时，RST 指令关闭指定的软元件。

当 SET 和 RST 命令均为 OFF 时，指定软元件的 ON/OFF 状态将不能改变。

[电路]



[时序图]

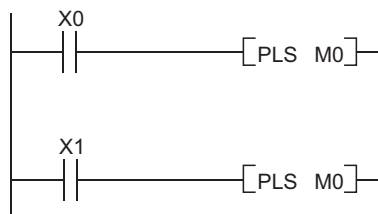


(3) 使用相同软元件的 PLS 指令

当 PLS 指令从关闭变为开启时, PLS 指令将指定的软元件开启。在其他任何时候均关闭指定的软元件 (OFF>OFF, ON>ON, 和 ON>OFF)。

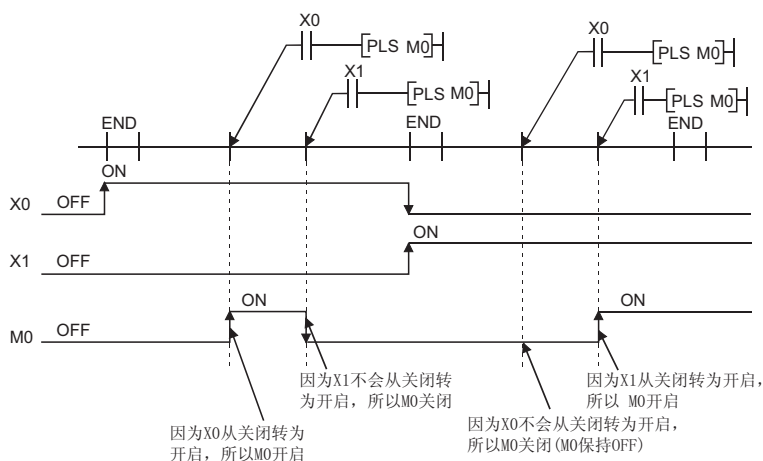
当两个或更多的使用相同软元件的 PLS 指令编在一个扫描周期中时, 每一个 PLS 指令从关闭变为开启时, 相对应的 PLS 指令将指定的软元件开启。在其他任何时候, 都将指定软元件设置为 OFF。

[电路]

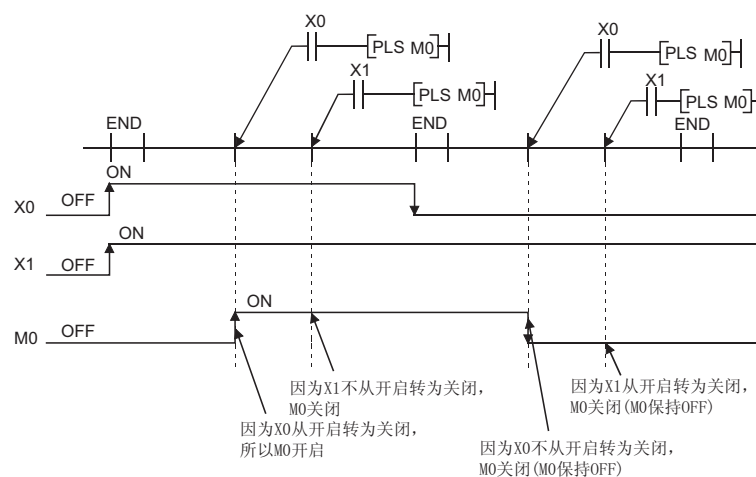


[时序图]

- X0 和 X1 的 ON/OFF 的时刻是不同的 (指定的软元件在整个扫描周期内不会开启)



- X0 和 X1 同时从关闭变为开启



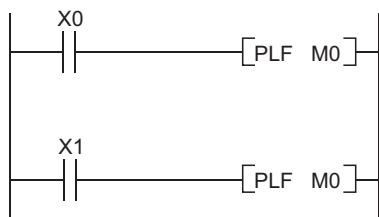
(4) 使用相同软元件的 PLF 指令

当 PLF 指令从关闭变成开启时，PLF 指令将指定的软元件开启，在其他任何时间里均将指定的软元件关闭 (OFF→OFF, OFF→ON, 和 ON→ON)。

如果在一个扫描周期中有两个或更多的使用相同的软元件的 PLF 指令时，每一个 PLF 指令在从开启变为关闭时，相应的 PLF 指令将指定的软元件关闭，并且在其他任何时间将指定的软元件将会关闭。

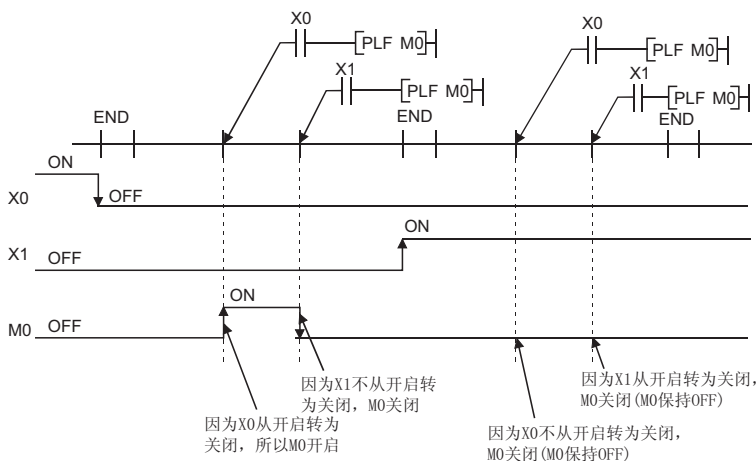
为此，当有两个或更多的使用相同软元件的 PLF 指令被编在一个扫描周期里时，在整个扫描周期内已经被开启的软元件不可以再次开启。

[电路]

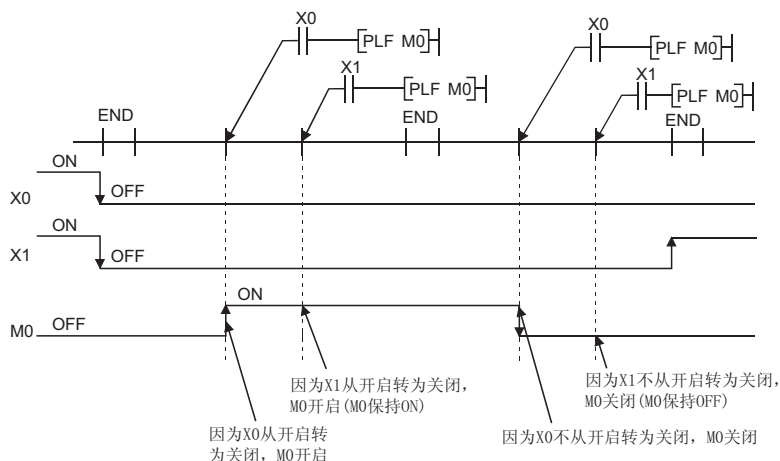


[时机图]

- X0 和 X1 ON/OFF 的时机是不同的（指定的软元件在整个扫描周期内不会开启）



- X0 和 X1 同时从开启变为关闭。



3. 10 使用文件寄存器的注意事项

本节将解释在 QCPU 和 QnACPU 里的使用文件寄存器的注意事项。

(1) 不能使用文件寄存器的 CPU 模块

Q00JCPU 不能使用文件寄存器。

当使用文件寄存器时，请使用 Q00JCPU 以外的 CPU 模块。

(2) 即将使用的文件寄存器的设定

如果要使用文件寄存器，即将使用的文件寄存器必须要用 PLC 参数或 QDRSET 指令来设定。（因为在“使用的文件寄存器”中已经预设，因此 Q00CPU 和 Q01CPU 的 PLC 参数不需要设定。）

如果即将使用的文件寄存器还没有设定，就不能使用该文件寄存器的指令执行普通的运行。

要点
如果即将使用的文件寄存器还没有在 PLC 参数中设定，那么可以新建一个使用文件寄存器的程序。当该程序写入到 CPU 模块并且被执行的时候，就不会出现错误。 但是，将正确的数据写入文件寄存器和从文件寄存器中读取正确数据都不能被执行。

(3) 文件寄存器区域的保护

(a) 高性能模型 QCPU，过程 CPU，QnACPU

当使用文件寄存器时，将文件寄存器登录到标准的 RAM/内存卡用以保护文件寄存器区域

(b) 基本类模型 QCPU (除 Q00JCPU)

文件寄存器区域已预先在标准的 RAM 中保护起来。

用户不需要再保护文件寄存器区域。

下列表格列出了可以在每一个 CPU 模块中使用文件寄存器的存储器。

存储器	QnACPU	高性能模型 QCPU, 过程 CPU	基本模型 QCPU (除 Q00JCPU)
标准 RAM	×	○	○
内存卡 *1, *2	○	○	×

○：能够被登录, ×: 不能被登录.

* 1: 当使用闪存时，只有从文件寄存器中读取能够被执行 (不能执行写入到闪存)

* 2: 当使用 E²PROM 时，写入到 E²PROM 可以通过 PROMWR 指令来执行。

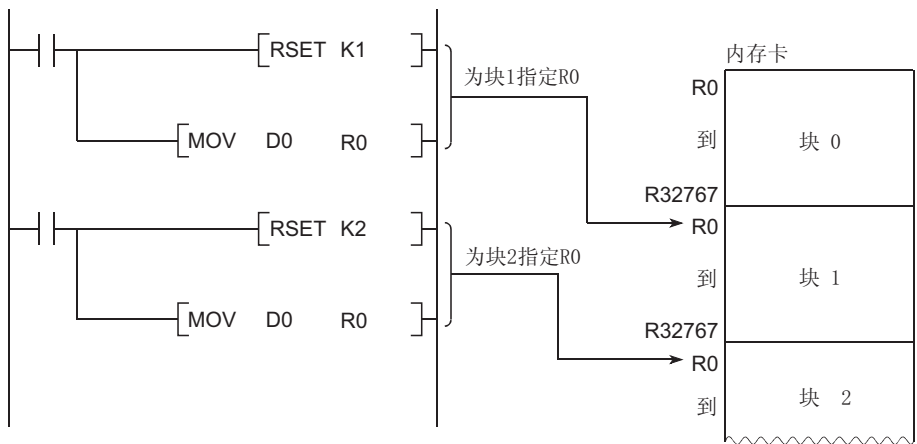
备注

关于文件寄存器的设定方法和文件寄存器区域的保护方法，请参考使用的 CPU 模块的用户手册 (功能解释，编程基础篇) 或 QnACPU 编程手册 (基础)。

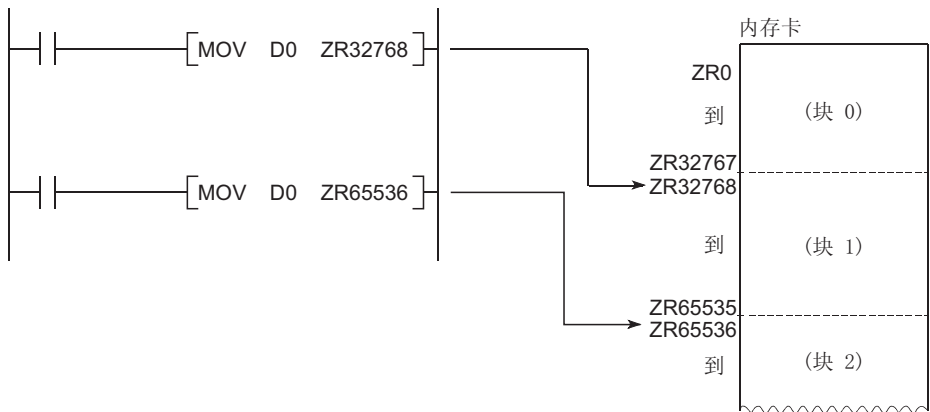
- (4) 文件寄存器号码的指定超过点数的指定数。
如果在文件寄存器中写入或读取的数据有比指定的点数更大的数，将不会出现错误。
然而，请注意在文件寄存器写入或读取正确的数据将不能被执行。

- (5) 文件寄存器指定方法
有块切换方法和序列号访问方法两种方法来指定文件寄存器。

- (a) 块切换方法
在块切换方法中，在 32k 点的单位（1 块）里指定使用的文件寄存器的点数。
对于 32k 点或更多的文件寄存器，通过切换 RSET 指令即将使用的块号来指定文件寄存器。
指定每一块为 R0 到 R32767。



- (b) 序列号访问方法
当使用序列号访问方法时，通过连续的软元件号指定超过 32k 点的文件寄存器。
多块的文件寄存能够当作连续的文件寄存器使用。
使用“ZR”为软元件名。



3.11 使用闪存的注意事项

本节介绍 QCPU 和 QnACPU 里的闪存使用的注意事项。

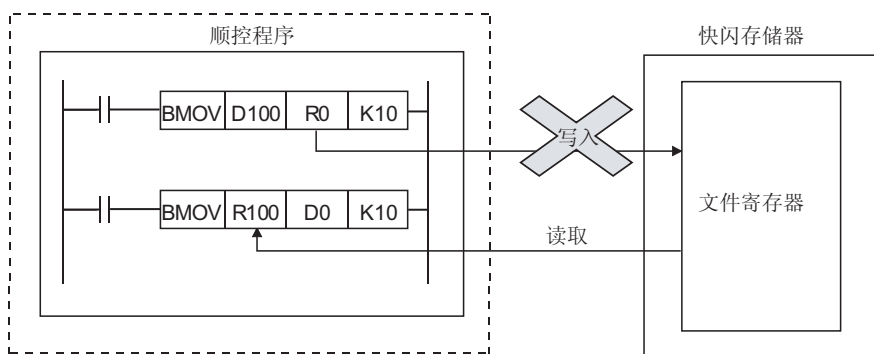
(1) 下列闪存可以和 QCPU 或 QnACPU 一起使用。

QCPU: 闪存卡

QnACPU: SRAM+闪存类型的内存卡

(2) 闪存只允许在顺控程序中读取。

(不能在顺控程序中执行对闪存的写入)。



如果要在闪存中存放文件寄存器，请先写入数据。

(a) QCPU

使用 GX Developer，将数据写入闪存卡。

(b) QnACPU

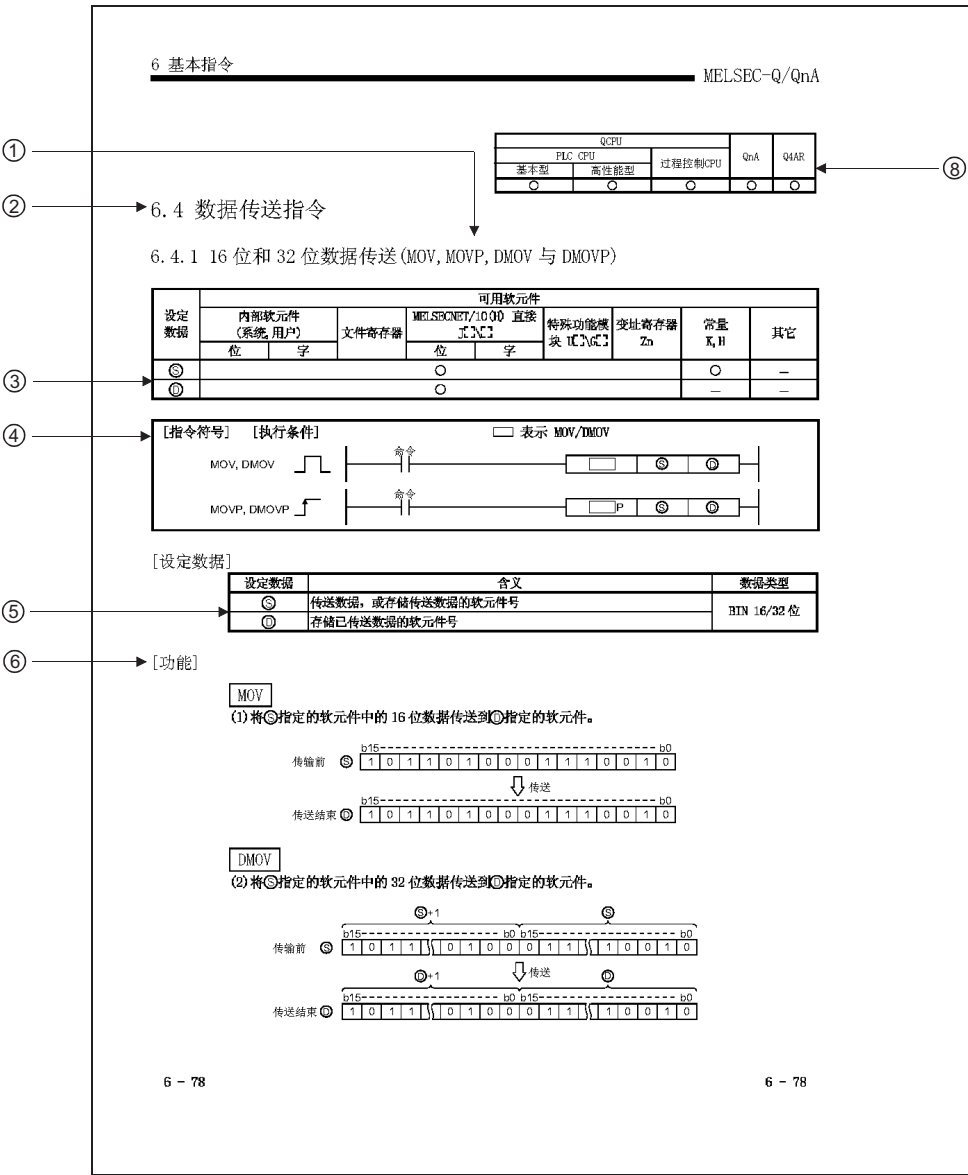
使用 IC 内存卡读取器/写入器，将数据写入到 SRAM+闪存型内存卡。

备忘录

[illegible]

4. 如何阅读指令

在后面各章节中的指令描述将以以下格式表述：



- ① 用来写指令的代码 (指令符号)。
 - ② 章节号和被讨论的指令的总分类。
 - ③ 被讨论的可以被指令使用的软元件用圆圈表示。
- 能被使用的软元件类型列示如下：

软元件类型	内部元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) *3 直接 J _□ K _□		特殊功能模块 U _□ G _□	索引寄存器 Z _n	常数 *1	其他 *1
	位	字		位	字				
可用 *4 软元件	X, Y, M, L, SM, F, V, B, SB, FX, FY *2	T, ST, C, *5 D, W, SD, SW, FD, @ _□	R, ZR	J _□ X J _□ Y J _□ B J _□ SB	J _□ W J _□ SW	U _□ G _□	Z	小数点常数 十六进制常数 实数常数 字符串常数	P, I, J, U, DX, DY, N, BL, TR, BL\ S, V

*1: 能够被设定的软元件被记录在“常数”和“其他”栏中。
*2: FX 和 FY 只能用于位数据, FD 只能用于字数据。
*3: 可以在 MELSECNET/H 和 MELSECNET/10 内使用。

6 基本指令

MELSEC-Q/QnA

[运行错误]

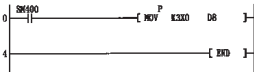
(1)不存在与MOV(P)或DMOV(P)指令相关的运行错误。

⑦

[程序示例]

(1)以下程序将来自X0到XB的输入数据存储在D8中。

[梯形图模式]

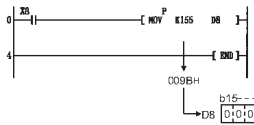


[列表模式]

步	指令	软元件
0	LD	S0400
1	MOV P	K155 D8
4	END	

(2)当XB变为ON时，以下程序将常数K155存储在D8中。

[梯形图模式]

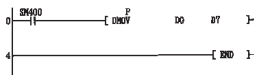


[列表模式]

步	指令	软元件
0	LD	X8
1	MOV P	K155 D8
4	END	

(3)以下程序将来自D0和D1的数据存储在D7和D8中。

[梯形图模式]

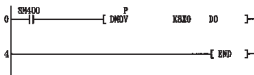


[列表模式]

步	指令	软元件
0	LD	S0400
1	DMOV P	D0 D7
4	END	

(4)以下程序将来自X0和X1F的数据存储在D0和D1中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	S0400
1	DMOV P	X0 D0
4	END	

※4：关于每一个软元件的详细资料，请参考使用的CPU的用户手册(功能解释，编程基础)或QnA编程手册。

※5：当T、ST和C被用于以下指令之外时，只有字数据可以使用。(位数据不能使用。)

[可以使用位数据的指令]

LD, LDI, AND, ANI, OR, ORI, LDP, LDF, ANDP, ANDF, ORP, ORF, OUT, RST

④ 表示梯形图模式的表达和指令的执行条件。

执行条件	非条件执行	在ON时执行	在ON时执行一次	在OFF时执行	在OFF时执行一次
代码记录在记述页	没有符号被记录				

⑤ 讨论为每个指令设定的数据和数据类型。

数据类型	含义
位	位数据或位数据中的第一个数字
BIN 16 位	BIN16 位数据或字软元件中的第一个数字。
BIN 32 位	BIN32 位数据或双字软元件中的第一个数字。
BCD 4 位	4 位 BCD 数据
BCD 8 位	8 位 BCD 数据
实数	浮动小数点数据
字符串	字符串数据
软元件名	软元件名称数据

⑥ 表示指令的功能。

⑦ 表示错误返回的条件及错误号。

⑧ 表示指令是否可以在每一种 CPU 模块类型中使用。

○: 可以被使用

△: 在限制情况 (功能版本, 软件版本) 下可以被使用

×: 不能被使用

⑨ 表示简单程序示例的梯形图和列表。

同样表示当程序被执行时, 使用的单个软元件的类型。

备忘录

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

5. 顺控程序指令

顺控程序指令包括继电器控制梯形图和相似情况下所用到的指令。可以分为以下种类：

指令	含义	参见
触点指令	运行开始，串行连接，并行连接	5.1 节
连接指令	梯形图块连接，从操作结果中生成脉冲，储存/读取操作结果	5.2 节
输出指令	位软元件输出，脉冲输出，输出反转	5.3 节
转移指令	位软元件转移	5.4 节
主控指令	主控	5.5 节
结束指令	程序结束	5.6 节
其他指令	程序停止，像无操作之类的不适合上述种类的指令	5.7 节

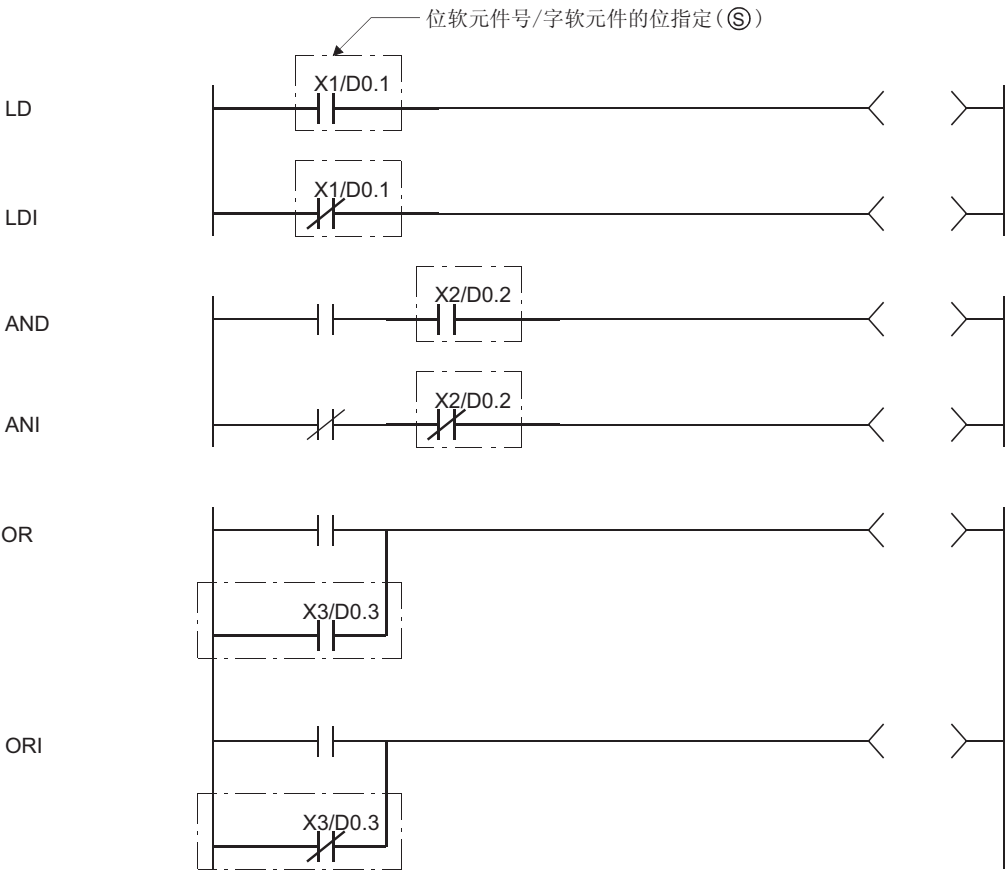
QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.1 触点指令

5.1.1 运行开始，串行连结，并行连接
(LD, LDI, AND, ANI, OR, ORI)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][]G[]	变址寄存器 Zn	常数 K, H	其他
	位	字		位	字				DX, BL
⑤	○					—		○	

[指令符号] [执行条件]



[设定数据]

设定数据	含义	数据类型
⑤	用作连接的软元件	位

[功能]

LD, LDI

(1) LD 是 A 触点操作开始指令, LDI 是 B 触点操作开始指令。它们从指定的软元件中读取 ON/OFF 信息, (如果一个字软元件位已经被指定, 这就成为了指定位的 1/0 状态), 并且将其作为一个操作结果。

AND, ANI

- (1) AND 是触点 A 串行连接指令，ANI 是触点 B 串行连接指令，它们读取指定位软元件的 ON/OFF 数据(如果字软元件已经完成了位指定，那么就读取指定位的 1/0 状态)，对该数据和到此点的操作结果执行 AND 操作，并将这个值作为运行结果。
- (2) 对 AND 或 ANI 的使用没有限制，但是下列内容适用于梯形图模式中的外围使用的软元件：
- (a) 写入..... 当 AND 和 ANI 为串行连接时，能够产生一个最大为 21 级的梯形图。
 - (b) 读取..... 当 AND 和 ANI 是串行连接时，能够显示一个最大为 24 级的梯形图。
如果数量超过 24 级，最大显示为 24 级。

OR, ORI

- (1) OR 是触点 A 单并行连接指令，ORI 是触点 B 单并行连接指令。它们从指定的软元件读取 ON/OFF 信息（如果字软元件位已经被指定，就变成指定位的 1/0 状态），并对到该点的运行结果执行 OR 运行，使用结果数值作为运行结果。
- (2) 对 OR 或 ORI 的使用没有限制，但是下列内容适用于梯形图模式中使用的外围设备：
- (a) 写入..... OR 和 ORI 能够创建最多 23 个梯形图的连接。
 - (b) 读取..... 最多可以显示 23 个用 OR 或 ORI 连接的梯形图。
第 24 个和后来的梯形图不能被正确显示。

备注

字软元件位的指定以十六进制形式进行设置

D0 的位 b11 应该是 D0.0B。

字软元件的位指定的更多信息，请参见 3.2.1 章节。

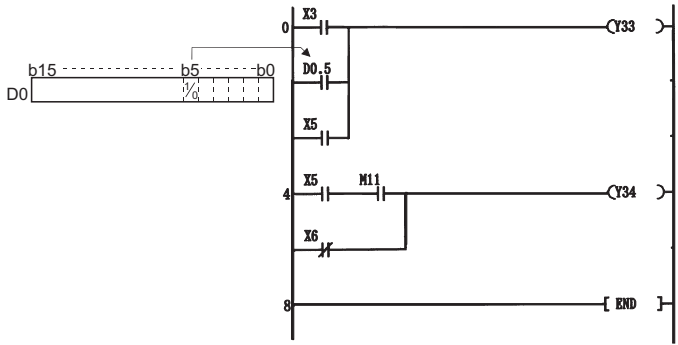
[运行错误]

(1) 在 LD, LDI, AND, ANI, OR, 或 ORI 指令下, 无运行错误。

[程序示例]

(1) 一个使用 LD, AND, OR, 和 ORI 指令的程序。

[梯形图模式]

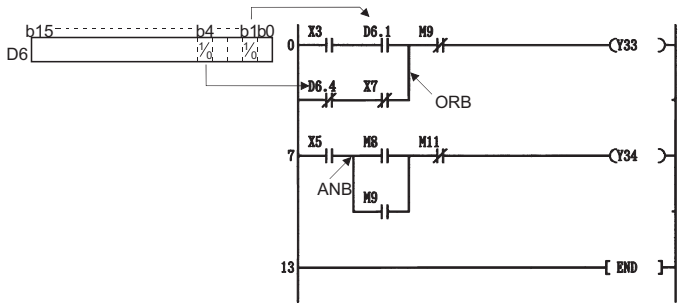


[列表模式]

步	指令	软元件
0	LD	X3
1	OR	D0.5... 字软元件
2	OR	X5
3	OUT	Y33
4	LD	X5
5	AND	M11
6	ORI	X6
7	OUT	Y34
8	END	

(2) 一个通过使用 ANB 和 ORB 指令建立起来的链接触点程序。

[梯形图模式]

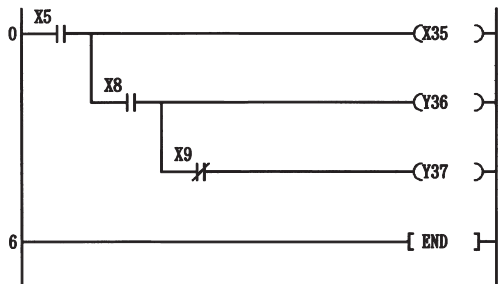


[列表模式]

步	指令	软元件
0	LD	X3
1	AND	D6.1... 字软元件
2	LDI	D6.4... 的位指定
3	ANI	X7
4	ORB	
5	ANI	M9
6	OUT	Y33
7	LD	X5
8	LD	M8
9	OR	M9
10	ANB	
11	ANI	M11
12	OUT	Y34
13	END	

(3) 一个使用 OUT 指令的并程序

[梯形图模式]



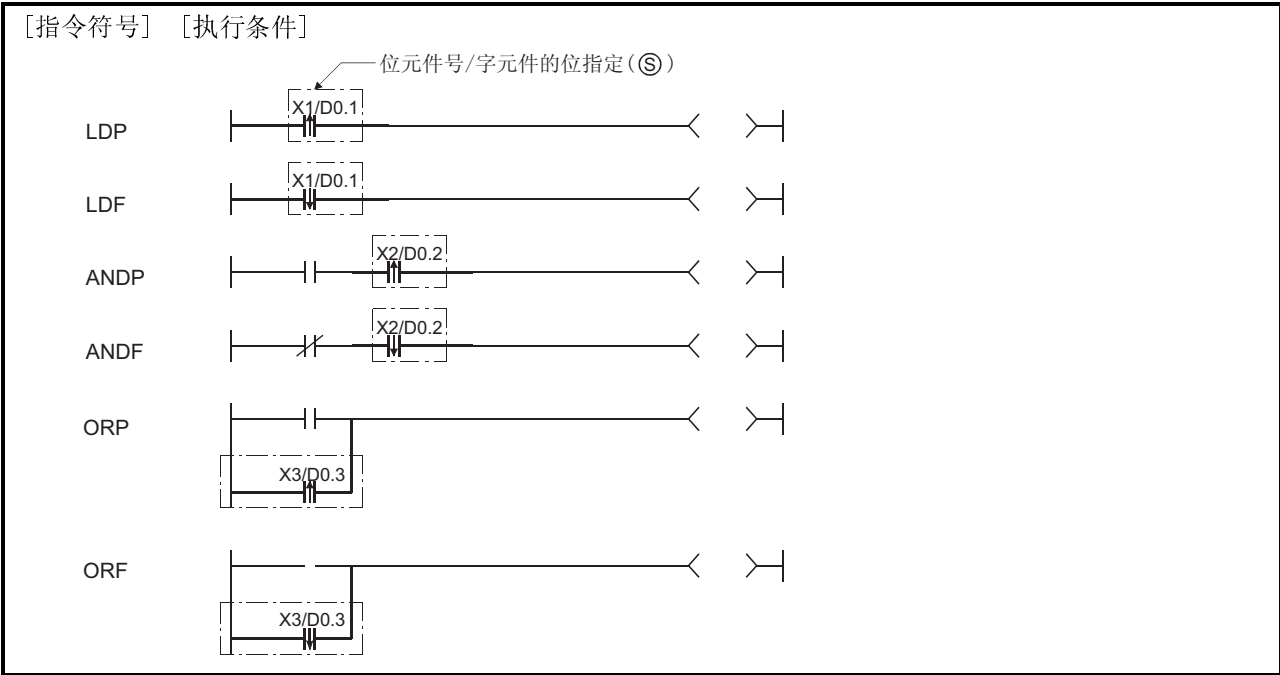
[列表模式]

步	指令	软元件
0	LD	X5
1	OUT	Y35
2	AND	X8
3	OUT	Y36
4	ANI	X9
5	OUT	Y37
6	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.1.2 脉冲运行开始，脉冲串行连接，脉冲并行连接
(LDP, LDF, ANDP, ANDF, ORP, ORF)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][] G[][]	变址寄存器 Zn	常数 K, H	其他
	位	字		位	字				DX
⑤	○					—		○	



[设定数据]

设定数据	含义	数据类型
⑤	用作触点的软元件	位

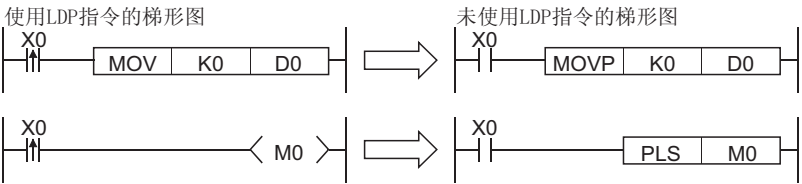
[功能]

LDP, LDF

(1) LDP 是前沿脉冲操作开始指令，并且只有在指定位软元件的前沿（当它从 ON 转换成 OFF）才为 ON。

如果字软元件已经被指定，仅当指定的位从 0 改变为 1 时才为 ON。

如果只有一个 LDP 指令，那么它和在 ON 时 ([] P) 执行的产生脉冲的指令具有同样作用。



- (2) LDF 是后沿脉冲操作开始指令，并且仅在指定位软元件的后沿（从 ON 转换成 OFF 时）才为 ON。
- 如果字软元件被指定，那么仅当指定的位从 1 变为 0 时才为 ON。

ANDP, ANDF

- (1) ANDP 是一个前沿脉冲串行连接指令，ANDF 是一个后沿脉冲串行连接指令。它们对至该点的运行结果执行 AND 操作，并且将结果值作为运行结果。
- 供 ANDP 和 ANDF 使用的 ON/OFF 数据，如下表显示：

被 ANDP 指定的软元件		ANDP 状态	被 ANDF 指定的软元件		ANDF 状态
位软元件	字软元件位指定		位软元件	字软元件位指定	
OFF → ON	0 → 1	ON	OFF → ON	0 → 1	OFF
OFF	0	OFF	OFF	0	
ON	1		ON	1	
ON → OFF	1 → 0		ON → OFF	1 → 0	ON

ORP, ORF

- (1) ORP 是一个前沿脉冲并行连接指令，ORF 是一个后沿脉冲并行连接指令，它们对至该点的运行结果执行 OR 操作，并且将结果值作为运算结果。

被 ORP 指定的软元件		ORP 状态	被 ORF 指定的软元件		ORF 状态
位软元件	字软元件位指定		位软元件	字软元件位指定	
OFF → ON	0 → 1	ON	OFF → ON	0 → 1	OFF
OFF	0	OFF	OFF	0	
ON	1		ON	1	
ON → OFF	1 → 0		ON → OFF	1 → 0	ON

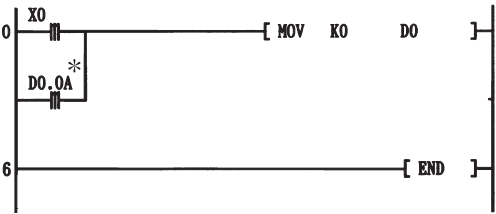
[运行错误]

- (1) 使用 LDP, LDF, ANDP, ANDF, ORP, 或 ORF, 不会有运行错误。

[程序示例]

- (1) 当在输入 X0 或者在数据寄存器 D0 的 b10（位 10）的前沿，下述程序执行 MOV 指令：

[梯形图模式]



[列表模式]

步	指令	元件
0	LDP	X0
2	ORP	D0.0A
4	MOV	K0
		D0
6	END	

备注

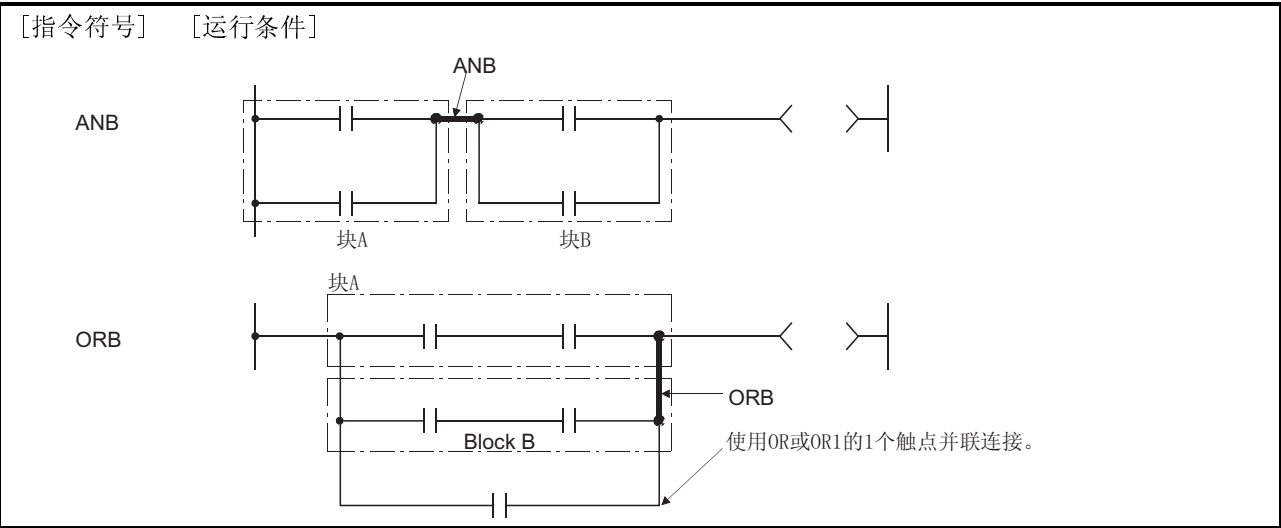
- *: 字软元件的位指定是以十六进制的形式执行的。
D0 的位 b10 应为 D0.0A。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.2 连接指令

5.2.1 梯形图块串行连接和并行连接(ANB, ORB)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][]G[][]	变址寄存器 Zn	常数 K, H	其他
	位	字		位	字				
—	—								



[功能]

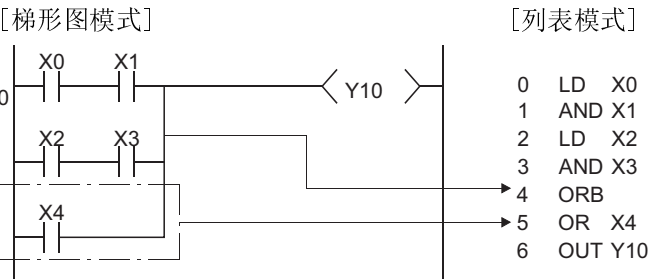
ANB

- (1) 在块 A 和块 B 上执行 AND 操作，并且将结果值作为运算结果。
- (2) ANB 的符号不是触点符号，而是连接符号。
- (3) 当在列表模式下编程时，最多可以连续写入 15 条 ANB 指令（16 块）。

ORB

- (1) 在块 A 和块 B 上执行一个 OR 操作，并且将结果值作为运行结果。

(2) ORB 是用于为有两个或更多触点的梯形图块执行并行连接的。
对于只有一个触点的梯形图块，使用 OR 或 ORI；在这种情况下没有必要使用 ORB。



(3) ORB 符号不是触点符号，而是连接符号。

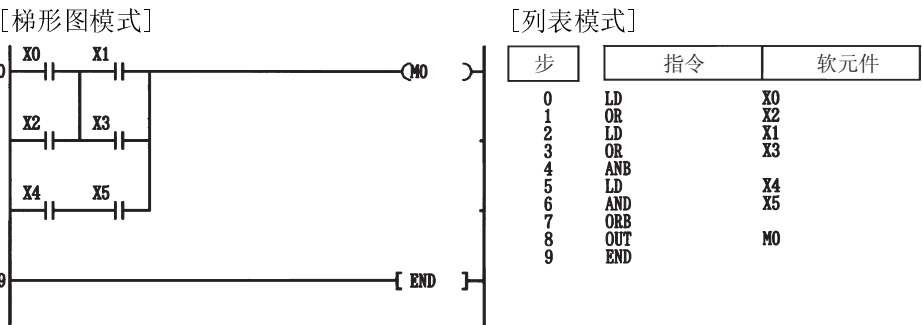
(4) 当在列表模式中编程时，可以连续使用最多 15 个 ORB 指令（16 块）。

[操作错误]

(1) 没有与 ANB 或 ORB 相关的操作错误。

[程序示例]

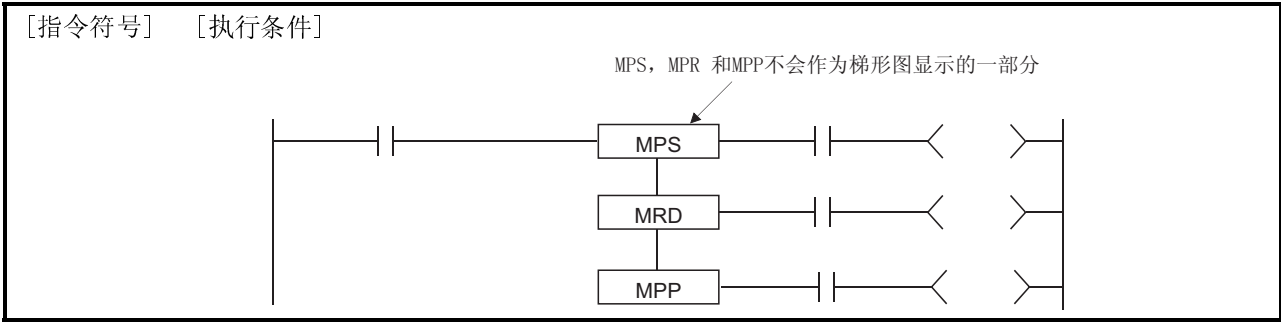
(1) 一个使用 ANB 和 ORB 指令的程序。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.2.2 运行结果推进，读取，弹出(MPS, MRD, MPP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常数 K, H	其他
	位	字		位	字				
—	—								



[功能]

MPS

- (1) 在MPS指令之前立即将运行结果（ON或OFF）储存在存储器内。
- (2) 可以连续使用最多16条MPS指令。
然而，在梯形图模式中最多能创建11条。
如果在该处理中使用了一个MPP指令，那么MPS指令计算使用的数量将会减少一个。

MRD

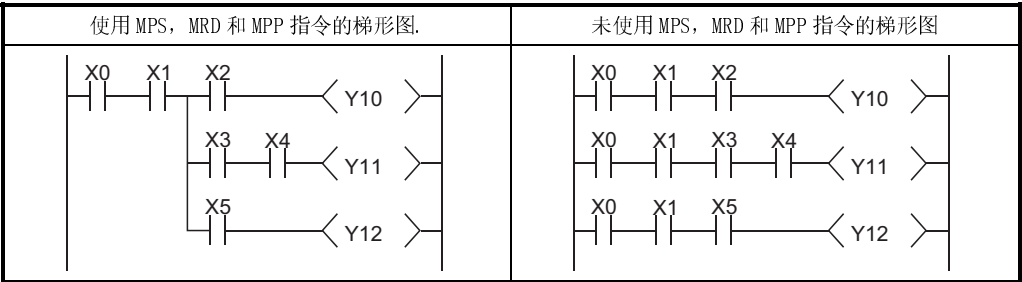
- (1) 读取为MPS指令储存的运算结果，并且使用该结果来执行下一步操作。

MPP

- (1) 读取为MPS指令储存的运算结果，并且使用该结果来执行下一步操作。
- (2) 清除被MPS指令储存的运算结果。
- (3) 从MPS指令使用次数数量中减去1。

要点

(1) 以下列出了已使用和未使用 MPS，MRD 和 MPP 指令的梯形图。



(2) MPS 和 MPP 指令必须使用同样的次数。

如果没有注意到这点，在外围设备中的梯形图模式中将会不会正确显示梯形图。

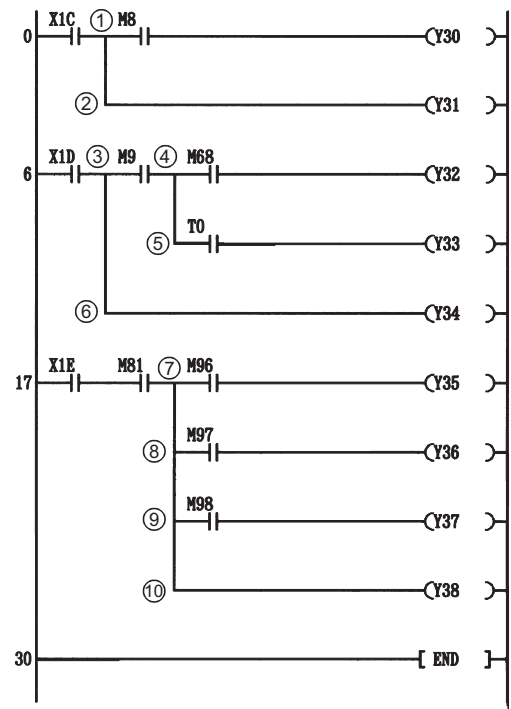
[操作错误]

(1) 没有与 MPS，MRD 或 MPP 指令相关的操作错误。

[程序示例]

(1) 一个使用 MPS，MRD 和 MPP 指令的程序。

[梯形图模式]

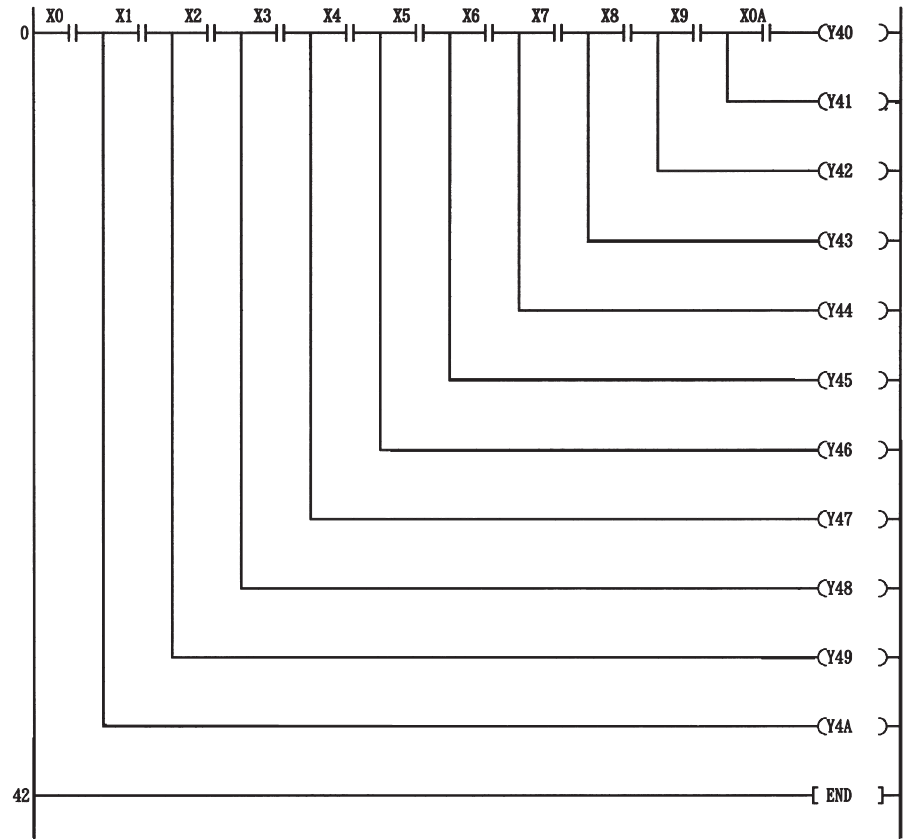


[列表模式]

步	指令	软元件
0	LD	X1C
1	MPS	M8
2	AND	M8
3	OUT	Y30
4	MPP	
5	OUT	Y31
6	LD	X1D
7	MPS	M9
8	AND	M9
9	MPS	M68
10	AND	M68
11	OUT	Y32
12	MPP	
13	AND	T0
14	OUT	Y33
15	MPP	
16	OUT	Y34
17	LD	X1E
18	AND	M81
19	MPS	M96
20	AND	M96
21	OUT	Y35
22	MRD	
23	AND	M97
24	OUT	Y36
25	MRD	
26	AND	M98
27	OUT	Y37
28	MPP	
29	OUT	Y38
30	END	

(2) 一个连续使用 MPS 和 MPP 指令的程序。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	MPS	
2	AND	X1
3	MPS	
4	AND	X2
5	MPS	
6	AND	X3
7	MPS	
8	AND	X4
9	MPS	
10	AND	X5
11	MPS	
12	AND	X6
13	MPS	
14	AND	X7
15	MPS	
16	AND	X8
17	MPS	
18	AND	X9
19	MPS	
20	AND	X0A
21	OUT	Y40
22	MPP	
23	OUT	Y41
24	MPP	
25	OUT	Y42
26	MPP	
27	OUT	Y43
28	MPP	
29	OUT	Y44
30	MPP	
31	OUT	Y45
32	MPP	
33	OUT	Y46
34	MPP	
35	OUT	Y47
36	MPP	
37	OUT	Y48
38	MPP	
39	OUT	Y49
40	MPP	
41	OUT	Y4A
42	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.2.3 运行结果反转（INV）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常数 K, H	其他
				位	字				U
	—	—							

[指令符号]	[执行条件]
INV	

[功能]

在 INV 指令之前立即转化运行结果。

在 INV 指令之前的直接运算结果	在 INV 指令执行之后的运算结果
OFF	ON
ON	OFF

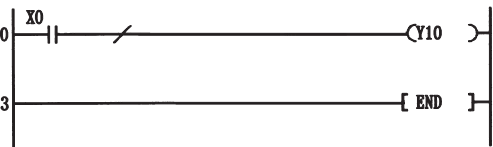
[操作错误]

(1) 没有与 INV 指令相关的操作错误。

[程序示例]

(1) 一个从 Y10 转换 X0 ON/OFF 数据和输出的程序。

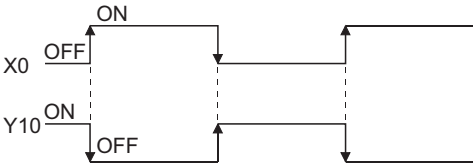
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	INV	
2	OUT	Y10
3	END	

[时序图]

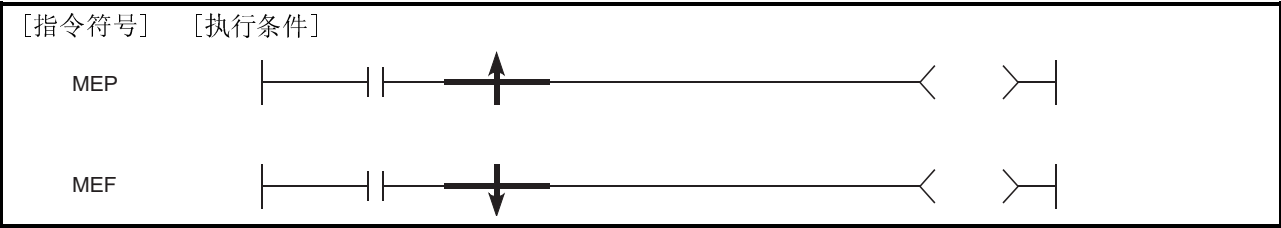


要点
(1) 建立在计算结果基础上的 INV 指令运行只有在 INV 指令给定的情况下才能进行。相应地，在 AND 指令相同的位置使用该指令，而 INV 指令不能在 LD 和 OR 的位置上使用。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.2.4 运算结果脉冲变换（MEP，MEF）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常数 K, H	其他 U
	位	字		位	字				
	—	—							



[功能]

- MEP
- (1) 如果一直到 MEP 指令前的运算结果都为前沿（从 OFF 到 ON），那么就为 ON（导通状态）。
如果一直到 MEP 指令前的运算结果都不为前沿，那么就为 OFF（非导通状态）。

(2) 当多触点串行连接时，MEP 指令的使用简化了脉冲变换处理。

- MEF
- (1) 如果一直到 MEF 指令前的运算结果都为后沿（从 ON 到 OFF），那么就为 ON（导通状态）。
如果一直到 MEF 指令前的运算结果都不为后沿，那么就为 OFF（非导通状态）。

(2) 当多触点串行连接时，MEF 指令的使用简化了脉冲变换处理。

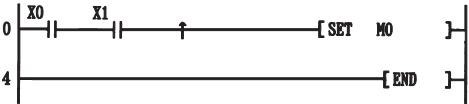
[操作错误]

- (1) 没有与 MEP 或 MEF 指令相关的操作错误。

[程序示例]

- (1) 在 X0 和 X1 的运算结果上执行脉冲变换的程序：

[梯形图模式]



[列表模式]

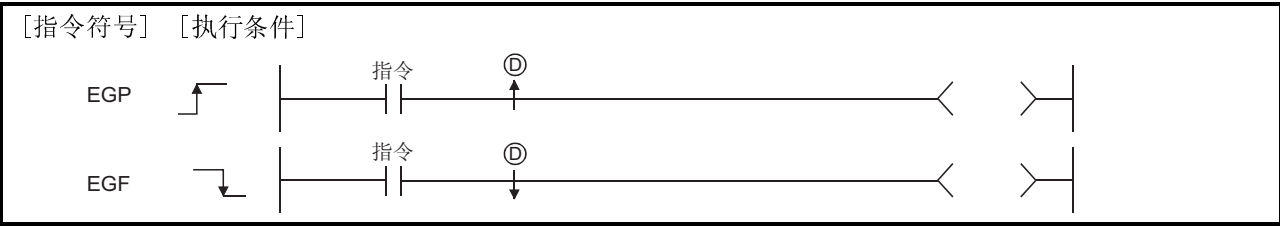
步	指令	软元件
0	LD	X0
1	AND	X1
2	MEP	
3	SET	MO
4	END	

要点	
(1)	当为一个已经被副程序或 FOR 到 NEXT 指令变址的触点进行脉冲变换时，MEP 和 MEF 指令偶尔会运行不正常。 如果进行脉冲变换的触点已经被副程序或 FOR 到 NEXT 指令索引过，可以使用 EGP/EGF 指令。
(2)	因为 MEP 和 MEF 指令是用在 MEP 和 MEF 指令之前的运算结果来进行操作的，因此在相同位置应该使用 AND 指令。 MEP 和 MEF 指令不能在 LD 或 OR 的位置使用。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.2.5 边沿继电器运算结果的脉冲变换 (EGF, EGF)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块U[][]G[][]	变址寄存器 Zn	常数 K, H	其他
	位	字		位	字				V
①	—								○



[设定数据]

设定数据	含义	数据类型
①	边沿继电器号运算结果储存的位置	位

[功能]

EGP

- (1) 执行 EGP 指令之前的运算结果被边沿继电器 (V) 储存在存储器里。
- (2) 执行 EGP 指令之前的运算结果在前沿 (OFF 到 ON) 为 ON (导通状态)。
如果执行 EGP 指令之前的运算结果不是在前沿 (例如, 从 ON 到 ON, 从 ON 到 OFF, 或从 OFF 到 OFF), 那么为 OFF (非导通状态)。
- (3) EGP 指令是在副程序里使用的, 并且用在 FOR 和 NEXT 指令之间通过索引修改对指定的程序进行脉冲操作。
- (4) EGP 指令可以同 AND 指令一样使用。

EGF

- (1) 执行 EGF 指令之前的运算结果被边沿寄存器 (V) 储存在存储器中。
- (2) 执行 EGF 指令之前的运算结果在后沿 (从 ON 到 OFF) 为 ON。
如果执行 EGF 指令之前的运算结果不是为后沿 (例如从 OFF 到 ON, 从 ON 到 ON, 或从 OFF 到 OFF), 则为 OFF (非导通状态)。
- (3) EGF 指令是在副程序里使用的, 并且用在 FOR 和 NEXT 指令之间通过索引修改对指定的程序进行脉冲操作。
- (4) EGF 指令可以同 AND 指令一样使用。

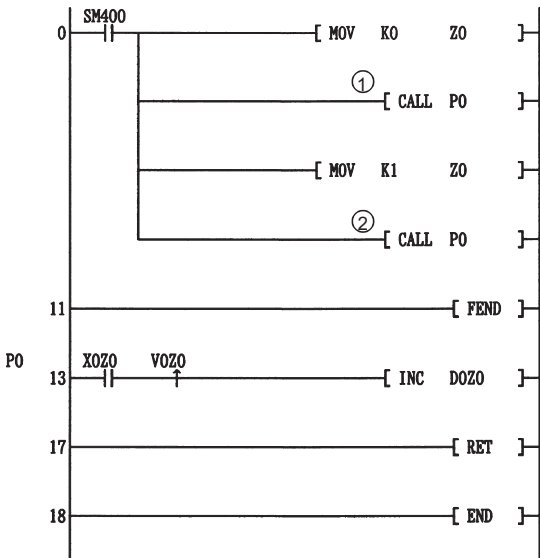
[操作错误]

(1) 没有与 EGP 或 EGF 指令有关的运行错误。

[程序示例]

(1) 包含一个使用 EGP 指令的副程序的指令。

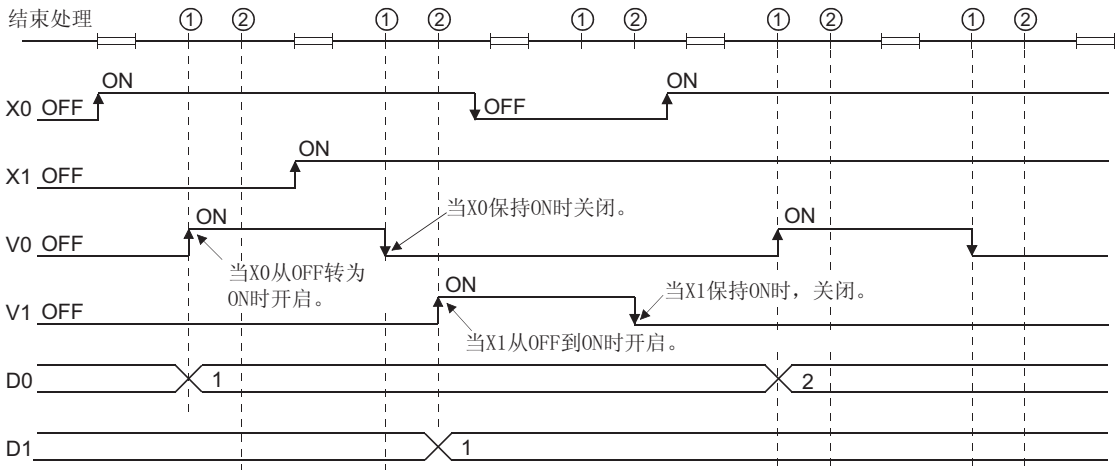
[梯形图模式]



[列表模式]

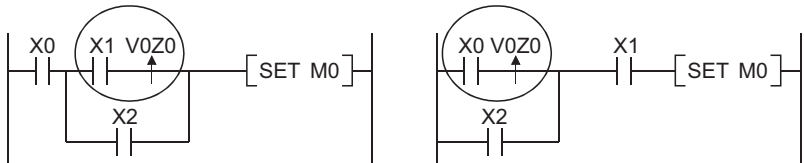
步	指令	软元件
0	LD	SM400
1	MOV	K0
		ZO
4	CALL	P0
6	MOV	K1
		ZO
9	CALL	P0
11	FEND	
12		P0
13	LD	X0Z0
14	EGP	VOZ0
15	INC	DOZO
17	RET	
18	END	

[操作]



要点

- (1) 因为 EGP 和 EGF 指令是按照在 EGP/EGF 指令之前的执行运算结果来执行的，因此这些指令必须在 AND 指令相同的位置上使用。（参见 5.1.1）
EGP 和 EGF 指令不能在 LD 或 OR 指令的位置上使用。
- (2) EGP 和 EGF 指令不能在下面所示的梯形图块位置使用。

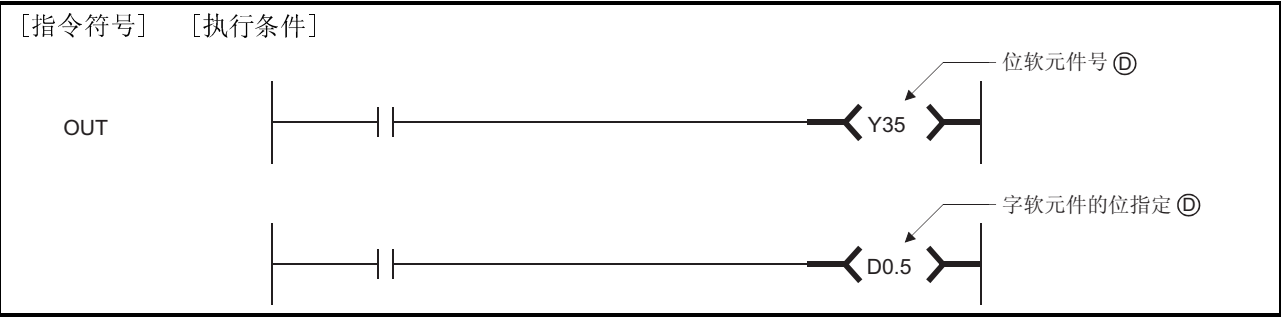


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.3 输出指令

5.3.1 输出指令（不包括定时器，计数器，和报警器）（OUT）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数 K, H	其他
	位	字		位	字				DY
①	○ (除了 T, C, 或 F)	○				—		○	



[设定数据]

设定数据	含义	数据类型
①	要开启和关闭的软元件号	位

[功能]

(1) 执行 OUT 指令之前的运算结果对于指定的软元件为输出。

运算结果	当使用位软元件时			当字软元件位指定已经完成
	线圈	触点		位指定
		A 触点	B 触点	
OFF	OFF	非导通	导通	0
ON	ON	导通	非导通	1

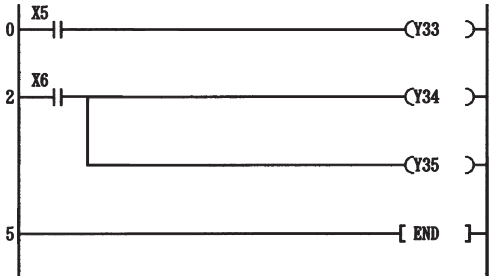
[运行错误]

(1) 在索引修改中的有关错误信息，参见章节 3.6。

[程序示例]

(1) 当位软元件被使用时

[梯形图模式]

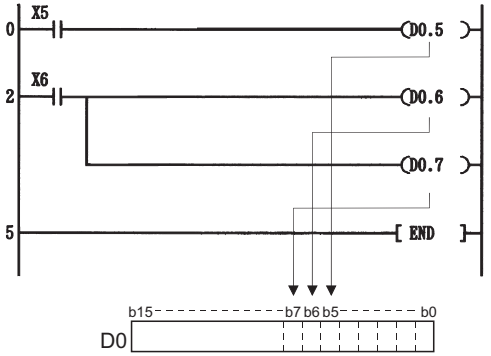


[列表模式]

步	指令	软元件
0	LD	X5
1	OUT	Y33
2	LD	X6
3	OUT	Y34
4	OUT	Y35
5	END	

(2) 当字软元件的位指定已经完成

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X5
1	OUT	D0.5
2	LD	X6
3	OUT	D0.6
4	OUT	D0.7
5	END	

备注

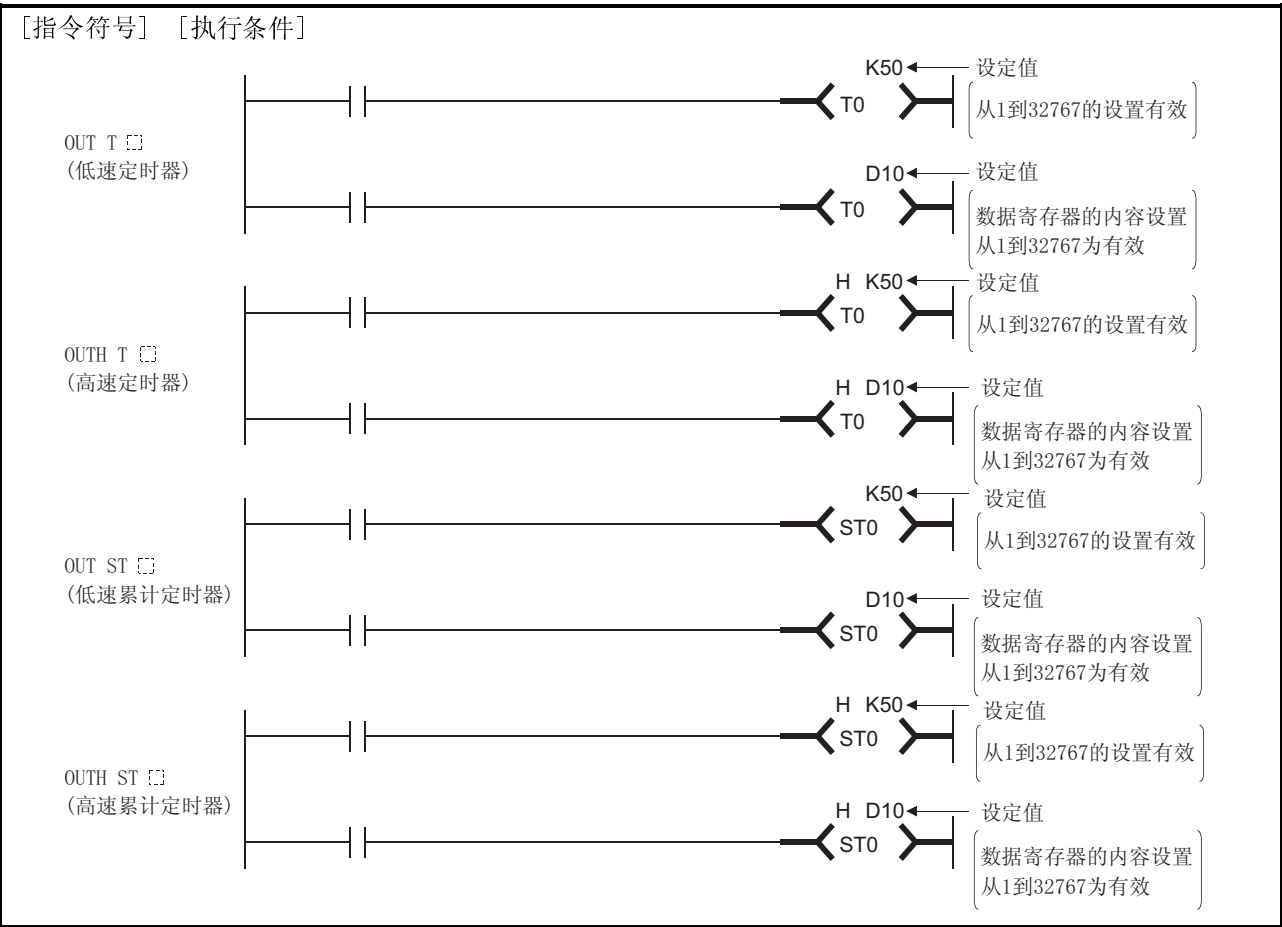
OUT 指令的基本步数如下：

- 当使用内部软元件或文件寄存器 (R) : 1
- 当使用直接访问输出 (DY) : 2
- 当使用任何其他软元件 : 3
(包括序列号访问文件寄存器)

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.3.2 定时器 (OUT T, OUTH T)

设定数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J□□□		特殊功能模 块 U□□G□	变址寄存器 Zn	常数 K	其他
	位	字		位	字				
⑤	○ (只有T)	—	—					—	—
设定值	—	○ (除了 T, C)	○ *2	—	○	—	—	○ *1	—

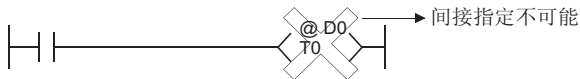


备注

- *1: 定时器的值只能设定为十进制常数 (K)。
定时器设定中不能使用十六进制 (H) 和实数
- *2: 在 Q00JCPU 中, 不能使用文件寄存器。

[设定数据]

设定数据	含义	数据类型
①	定时器号	位
设定值	定时器设置值	16 位 *3

要点
<p>*3: 定时器值的设置不能间接指定。</p>  <p>关于间接指定的更多信息，参见章节 3.4</p>

[功能]

(1) 如果执行 OUT 指令之前的运算结果是 ON，那么定时器线圈为 ON 并且定时器计数到设定值；当到达时间到状态（总数值等于或比设定值大）时，触点应答如下：

A 触点	导通
B 触点	非导通

(2) 当到 OUT 指令的运算结果是从 ON 变换到 OFF 时，触点应答如下：

定时器类型	定时器线圈	定时器当前值	在时间到之前		时间到之后	
			A 触点	B 触点	A 触点	B 触点
低速定时器	OFF	0	非导通	导通	非导通	导通
高速定时器						
低速累计定时器	OFF	保持当前值	非导通	导通	导通	非导通
高速累计定时器						

* 在低速累计定时器和高速累计定时器中，通过使用 RST 指令，当前值被清除，并且触点被复位。

(3) 通过使用 RST 指令在时间到之后，清除累计定时器的当前值并且将触点关闭。

(4) 负数（-32768 至 -1）不能在定时器中被设定为设定值。
如果设定值为 0，当时间 OUT 指令被执行时，定时器将会超时。

(5) 当 OUT 指令被执行时，下列处理会进行：

- OUT T □ 线圈变为 ON 或 OFF
- OUT T □ 触点变为 ON 或 OFF
- OUT T □ 当前值更新

如果在 OUT T □ 指令为 ON，JMP 指令或类似的指令被用来跳至 OUT T □ 指令时，不会有任何当前值更新或者进行触点 ON/OFF 操作。同样，如果在同一个扫描周期内同一 OUT T □ 指令被执行了两次或更多次，那么执行的重复次数的当前值将会被更新。

(6) 定时器的索引修改只能被 Z0 或 Z1 执行。
对于定时器的设定值不能进行索引修改。

备注

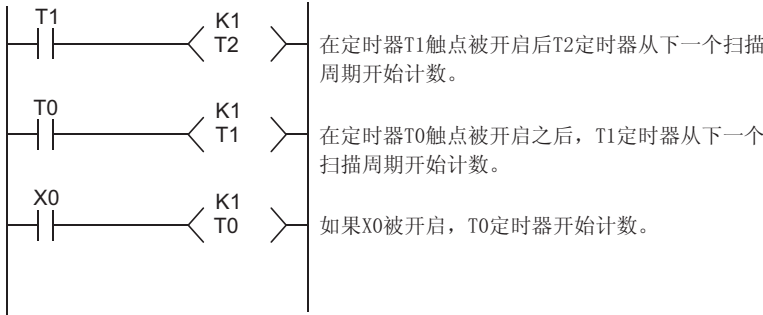
- (1) 低速定时器的默认值和低速累计时间定时器的极限为 100 ms。低速定时器的默认值和低速累计时间定时器的极限可以在参数模式“PLC 系统设定”区域内设定，在 10 ms 至 1 秒之间的时限上增加 10 ms。
- (2) 高速定时器的默认值和高速累计时间定时器时限为 10 ms。
高速定时器的默认值和高速累计时间定时器的时限可以在参数模式“PLC 系统设定”区域内设定，在 10 ms 至 100 ms 之间的时限上增加 1ms。
- (3) 关于定时器的计数方法，请参考所用 CPU 模块的用户手册（功能解释，编程基础）或 QnACPU 编程手册。（基础级）

[注意事项]

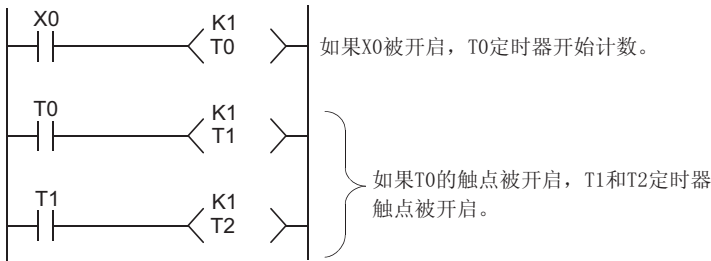
- (1) 当创建的程序中定时器触点触发其他的定时器的操作，按照定时器的操作顺序创建程序-即为后操作的定时器先创建程序。
在下列情况中，如果程序是按照定时器操作的顺序来创建的，那么在相同的扫描周期内，所有的定时器均为 ON。
 - 如果设定值小于一个扫描周期。
 - 如果“1”被设定。

示例

- 对于 T0 至 T2 的定时器，按照定时器后操作的顺序创建程序。



- 对于定时器 T0 至 T2，程序按照定时器操作的顺序创建。



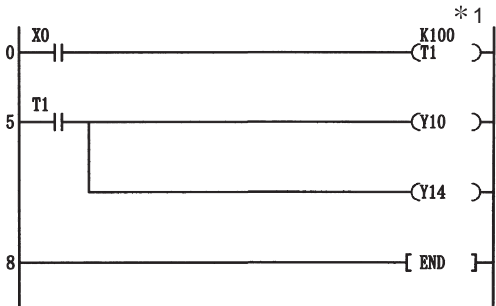
[运行错误]

- (1) 没有与 OUT T []指令相关的运行错误。

[程序示例]

(1) 下列程序在 X0 已经接通后将 Y10 和 Y14 开启 10 秒。

[梯形图模式]

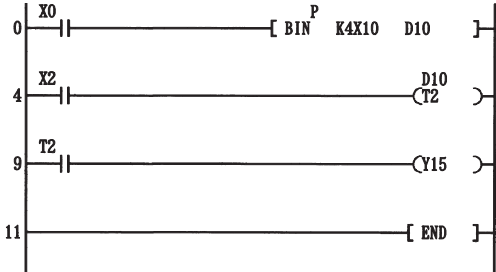


[列表模式]

步	指令	软元件
0	LD	X0
1	OUT	T1 K100
5	LD	T1
6	OUT	Y10
7	OUT	Y14
8	END	

(2) 下列程序使用从 X10 至 X1F 的 BCD 数据作为定时器的设定值。

[梯形图模式]



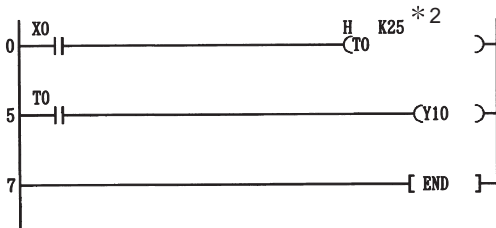
将从X10至X1F的BCD数据转换成BIN并且存储在D10。
当X2接通，存储在D10的数据被计算为设置值。
当T2计数完时Y15接通。

[列表模式]

步	指令	软元件
0	LD	X0
1	BINP	K4X10 D10
4	LD	X2
5	OUT	T2 D10
9	LD	T2
10	OUT	Y15
11	END	

(3) 下列程序在 X0 接通后将 Y10 开启 250ms

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	OUTH	T0 K25
5	LD	T0
6	OUT	Y10
7	END	

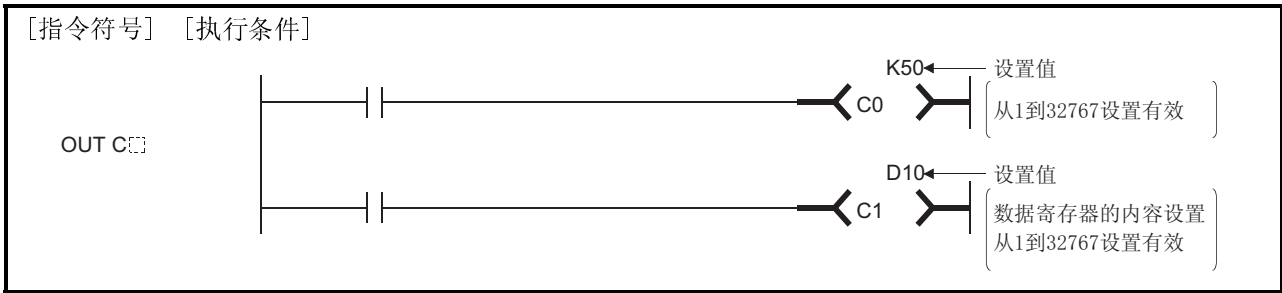
备注

- (1) *1: 低速定时器的设定值表示它的默认时限 (100ms)
- (2) *2: 高速定时器的设定值显示表示其默认时限为 (10ms)
- (3) OUT T □ 指令的基本步数为 4。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.3.3 计数器 (OUT C)

设定数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常数 K	其他
	位	字		位	字				U
⑤	○ (只有 C)	—	—				—	—	
设定值	—	○ (除了 T, C)	○ *3	—	○	—	○ *1	—	



[设定数据]

设定数据	含义	数据类型
①	计数器号	位
设定值	计数器设定值	BIN 16 位 *2

[功能]

- (1) 如果执行 OUT 指令之前的运算结果从 OFF 变为 ON, 在当前值 (计数值) 上要加上 1, 并且计数完成 (当前值=设定值) 时, 触点应答如下:

A 触点	导通
B 触点	非导通

要点

*2: 计数器值不能通过间接指定来设定

关于间接指定的更多信息, 请参见章节 3.4

备注

- 关于计数器计数方法, 请参见所使用的 CPU 模块使用手册 (功能解释, 编程基础) 或 QnACPU 编程手册 (基础)
- *1: 计数器值只能设定为十进制常数 (K)。
计数器值设定不能使用十六进制常数 (H) 或实数。

- (2) 在运算结果为 ON 时，不会进行任何计数。
(对于计数输出，没有必要进行脉冲变换)
- (3) 如果达到计数值后，在 RST 指令执行之前，不改变计数值或触点
- (4) 定时器中设定值不能为负数（-32768 至 -1）
如果设定值是 0，那么和 1 所进行的处理是一样的
- (5) 计数器线圈和触点的索引修改只能使用 Z0 和 Z1
不能为定时器设定执行索引修改。

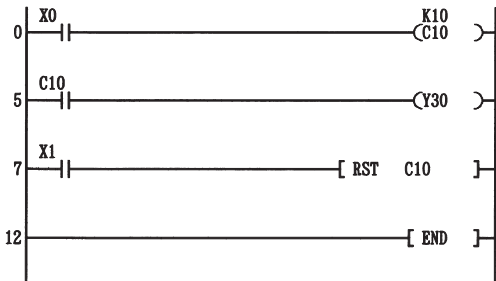
[运行错误]

- (1) 没有与 OUT C []指令相关的运行错误。

[程序示例]

- (1) 下列程序在 X0 接通 10 次后将 Y30 开启，当 X1 接通的时候复位计数器。

[梯形图模式]

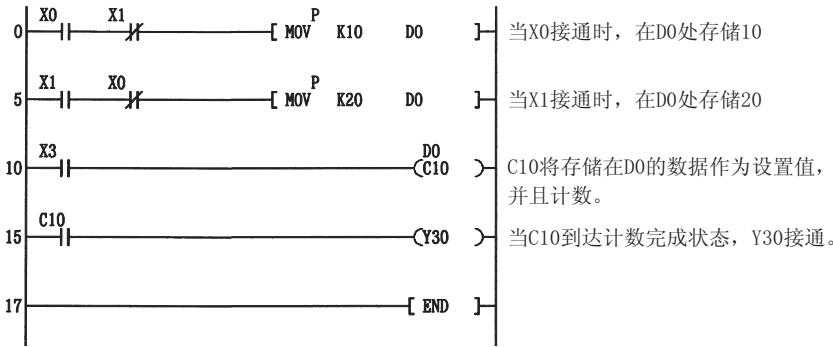


[列表模式]

步	指令	软元件
0	LD	X0
1	OUT	C10
		K10
5	LD	C10
6	OUT	Y30
7	LD	X1
8	RST	C10
12	END	

- (2) 下列程序当 X0 接通时在 10 的位置设定 C10 的值，当 X1 接通时在 20 的位置设定 C10 的值。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	ANI	X1
2	MOVP	K10
		D0
5	LD	X1
6	ANI	X0
7	MOVP	K20
		D0
10	LD	X3
11	OUT	C10
		D0
15	LD	C10
16	OUT	Y30
17	END	

备注

OUT C [] 指令的基本步数为 4。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.3.4 报警器输出 (OUT F)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
①	○(只有 F)						



[设定数据]

设定数据	含义	数据类型
①	要开启的报警器号	位

[功能]

- 执行 OUT 指令之前的运算结果被输出到指定的报警器。
- 当报警器 (F) 开启的时候, 出现下列应答:
 - [使用 Q3A, Q4A, 或 Q4ARCPU]
 - 报警器号显示在 CPU 模块前面的 LED 显示软元件上, 并且 “USER” LED 接通。
 - 为 ON 的报警器号 (F 号) 储存在特殊寄存器里 (SD64 至 SD79)
 - SD63 的值增加 1
 - [使用上述之外的 CPU 模块]
 - “USER” LED 接通。
 - 为 ON 的报警器号 (F 号) 储存在特殊寄存器中 (SD64 至 SD79)
 - SD63 的值增加 1。
- 如果 SD63 的值是 16 (当有 16 个报警器已经为 ON 时出现这种情况), 即使有一个新的报警器被开启, 它的号也不会储存在 SD64 至 SD79 中。

(4) 当报警器被 OUT 指令关闭时，会出现下列反应。

[使用 Q3A, Q4A, 或 Q4ARCPU]

- 线圈断开，但是在 CPU 模块前面的 LED 显示软元件，“USER” LED 状态，以及储存在 SD63 至 SD79 值的内容不会有改变。
- 使用 RST F 指令来关闭在 CPU 模块前面的 LED 显示软元件和“USER” LED，并且在 SD63 至 SD79 上删除被 OUT F 指令关闭的报警器。

[使用上述之外的 CPU]

- 线圈断开，但是在“USER” / “错误” LED 状态和储存在 SD63 至 SD79 的值的内容不会有变更。
- 使用 RST F 指令来关闭“USER” / “错误” LED 并且在 SD63 至 SD79 上删除被 OUT F 指令关闭的报警器。

[运行错误]

(1) 没有与 OUT F 指令有关的运行错误。

备注

(1) 关于报警器的详细资料，请参见使用的 CPU 模块用户手册（功能解释，编程基础）或 QnACPU 编程手册（基础）

(2) OUT 模块 F 指令的基本步号为 4。

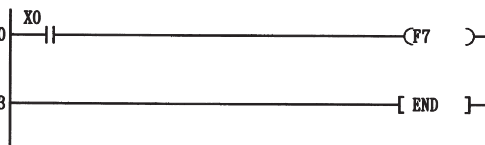
(3) 下列表格显示哪一个 CPU 模块有在 CPU 模块正面的 LED 显示软元件或“USER” LED。

LED 类型	CPU 模块类型名称
LED 显示软元件	Q3A, Q4A, Q4AR
“USER” LED	Q2A(S1), Q2AS(S1), Q2ASH(S1), 高性能模型 QCPU
“ERR” LED	基本模型 QCPU

[程序示例]

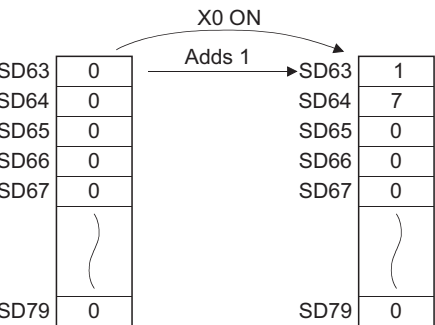
(1) 下列程序在 X0 接通的时候开启 F7，并且在从 SD64 至 SD79 上存储值 7。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	OUT	F7
3	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.3.5 设定软元件（报警器除外）（SET）

设定 数据	可用软元件									
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][]G[][]	变址寄存器 Zn	常数 K, H	其他	
	位	字		位	字				BL	DY
⑤	○	○ (除了 T, C)	○	○		—	—	○	○	



[设定数据]

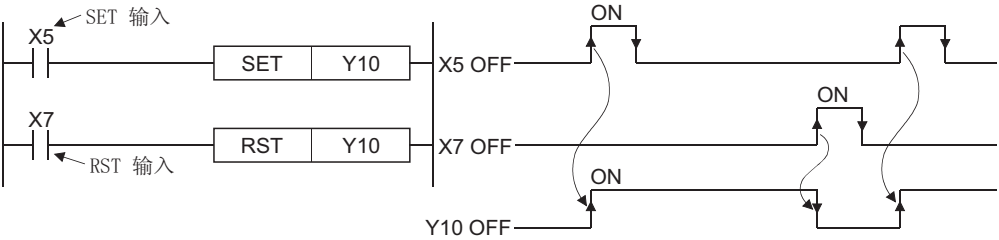
设定数据	含义	数据类型
⑤	待设定的位软元件号 (ON) / 字软元件的位指定	位

[功能]

(1) 当 SET 输入为 ON 时，指定的软元件应答如下：

软元件	软元件状态
位软元件	线圈和常数开启
当字软元件的位指定已经完成	指定位置为 1

(2) 即使 SET 输入变为 OFF 时，开启的软元件将仍然保持 ON 的状态。
被 SET 指令开启的软元件能够被 RST 指令关闭。



(3) 当 SET 输出为 OFF 时，软元件状态不会改变。

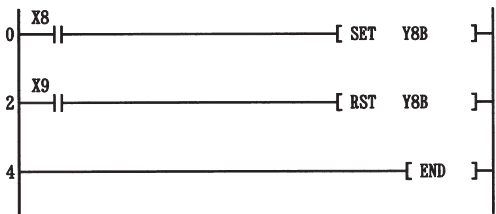
[运行错误]

(1) 没有与 SET 指令相关的运行错误。

[程序示例]

(1) 下列程序在 X8 接通时设定 Y8B (ON)，并且在 X9 接通的时候重置 Y8B (OFF)

[梯形图模式]

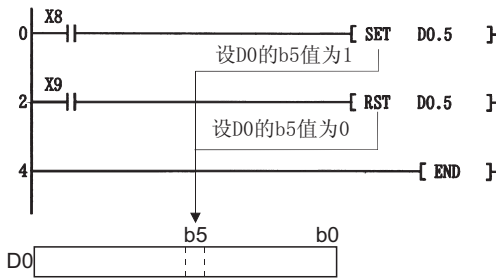


[列表模式]

步	指令	软元件
0	LD	X8
1	SET	Y8B
2	LD	X9
3	RST	Y8B
4	END	

(2) 下列程序在 X8 接通时设定 D0 位 5 (b5) 的值至 1，并且在 X9 接通时，设定位值至 0。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X8
1	SET	D0.5
2	LD	X9
3	RST	D0.5
4	END	

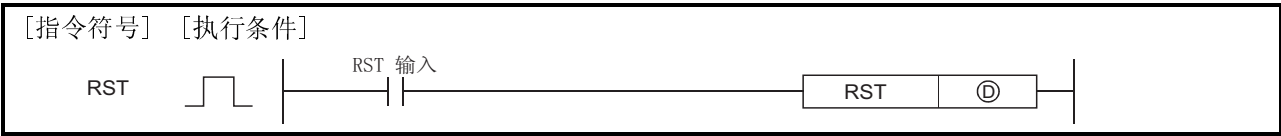
备注

- 基本的 SET 指令如下
 - 当内部软元件或文件寄存器 (R0 至 R32767) 在使用 : 步 1
 - 当直接访问输出 (DY) 或 SRC 程序软元件 (BL) 在使用 : 步 2
 - 当一些其他的软元件在使用 : 步 3
- 当 X 被当做软元件使用时，使用没有被用做实际输入的软元件号。如果在实际输入软元件和输入 X 中使用同一个号，那么实际输入的数据将会被在 SET 指令中指定的输入 X 覆盖。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.3.6 复位软元件（除了报警器）（RST）

设定 数据	可用软元件									
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J K L		特殊功能模 块 U V G	变址寄存器 Zn	常数 K, H	其他	
	位	字		位	字				BL	DY
④	○	○	○	○		○	—	—	○	



[设定数据]

设定数据	含义	数据类型
④	待设定的位软元件号/字软元件的位指定	位
	待复位的字软元件号	BIN 16 位

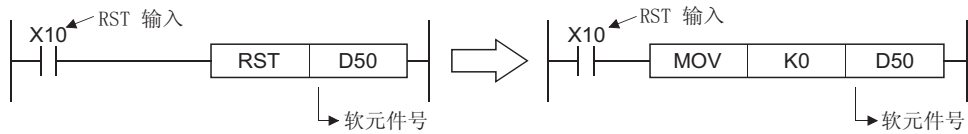
[功能]

(1) 当 RST 输入开启时指定的软元件应答如下：

软元件	软元件状态
位软元件	将线圈和触点关闭
定时器和计数器	将当前值设定位 0，并且将线圈和触点关闭。
当字软元件的位指定已经完成	设定指定位的值到 0
除定时器和计数器之外的字软元件	将触点设置到 0

(2) 当 RST 输入为 OFF，软元件状态不改变。

(3) 被 RST 指令指定的字软元件功能等同于下列梯形图



[运行错误]

(1) 没有与 RST 指令相关的运行错误。

备注

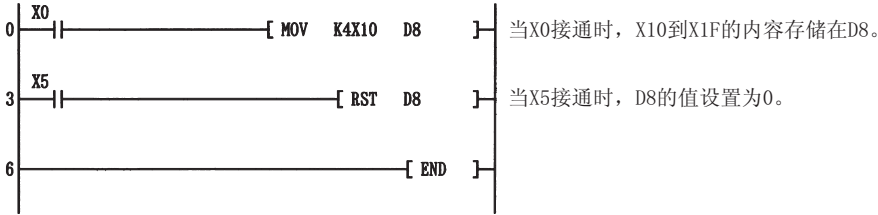
- 1) RST 指令的基本步数如下：
 - a) 对于位处理
 - 内部软元件（有待被位软元件或字软元件指定的位）：1
 - 直接输出：2
 - 定时器，计数器：4
 - 上述之外的其他：3

- b) 对于字处理
- 内部软元件（有待被位软元件指定的字） : 2
 - 变址寄存器 : 2
 - 上述之外的其他 : 3

[程序示例]

(1) 下列程序将数据寄存器的值设定为 0

[梯形图模式]

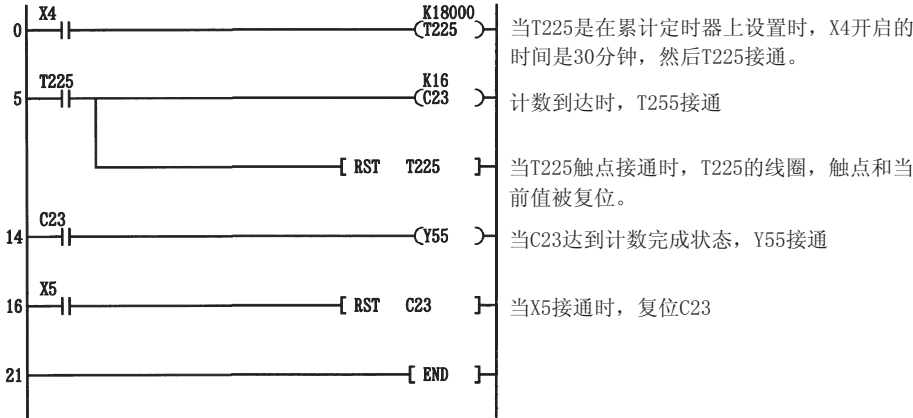


[列表模式]

步	指令	软元件
0	LD	X0
1	MOV	K4X10 D8
3	LD	X5
4	RST	D8
6	END	

(2) 下列程序复位 100ms 累计定时器 and 计数器

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X4
1	OUT	T225 K18000
5	LD	T225
6	OUT	C23 K16
10	RST	T225
14	LD	C23
15	OUT	Y55
16	LD	X5
17	RST	C23
21	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.3.7 设定和复位报警器 (SET F, RST F)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常数 K, H	其他
	位	字		位	字				U
①	○ (只有 F)	—							



[设定数据]

指令名称	设定数据	含义	数据类型
SET	①	待设定的报警器号 (F 号)	位
RST	①	待复位的报警器号 (F 号)	

[功能]

SET

- 当 SET 输入接通时被①指定的报警器接通。
- 当报警器 (F) 接通时出现下列应答：
 - [使用 Q3A, Q4A 或 Q4ARCPU]
 - 报警器号显示在 CPU 模块前面的 LED 显示软元件上，或 “USER” LED 亮。
 - 开启的报警器号 (F 号) 连续地存储在特殊寄存器里 (SD63 至 SD79)
 - SD63 的值增加 1
 - [使用 Q2A (S1), Q2AS (S1), Q2ASH (S1) 或 QCPU]
 - “USER” LED 亮。
 - 开启的告警器号 (F 号) 被连续地存储在特殊寄存器里 (SD63 至 SD79)。
 - SD63 的值增加 1。
- 如果 SD63 的值为 16 (当有 16 个报警器已经为 ON 时出现这种情况)，即使有新的报警器被开启，它的号也不会被 SD64 至 SD79 存储。

RST

- 当 RST 输入接通时，被①指定的报警器被关闭。
- 已经断开的报警器的报警器号从特殊寄存器上 (SD64 至 SD79) 删除，并且 SD63 的值减小 1。

备注

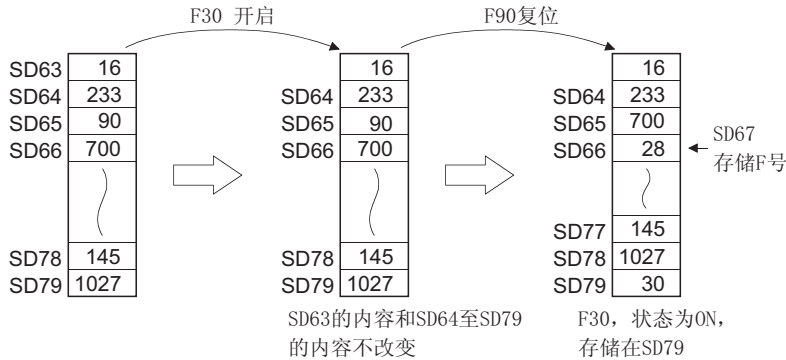
- 关于报警器的详细资料，请参见所用的 CPU 模块的用户手册 (功能解释，编程基础) 或 QnACPU 编程手册 (基础)

(2) 下表列出了在 CPU 模块正面有 LED 显示软元件或具有“USER”LED 的模块。

LED 类型	CPU 模块类型名称
LED 显示软元件	Q3A, Q4A, Q4AR
“USER”LED	Q2A(S1), Q2AS(S1), Q2ASH(S1), QCPU

(3) 如果，当 SD63 的值为 16 时，通过使用 RST 指令将报警器号从 SD64 至 SD79 上删除，报警器号没有在 SD64 至 SD79 上登录的报警器将会开启，这些报警器号也将被登录。如果所有 SD64 至 SD79 上的报警器号都被关闭，在 CPU 模块前端的 LED 显示软元件，或“USER”LED，将会被关闭。

[当 SD63 为 16 时将会发生的操作]



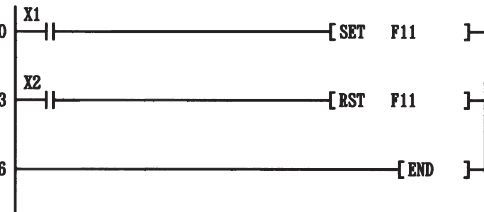
[运行错误]

(1) 没有与 SET F [] 或 RST F [] 指令相关的运行错误。

[程序示例]

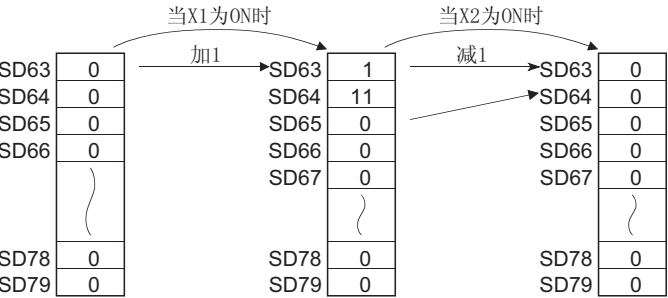
(1) 下列程序在 X1 接通时将报警器 F11 开启，并且在特殊寄存器（SD64 至 SD79）上存储值 11。然后，当 X2 接通时将报警器 F11 复位，并且从特殊寄存器（SD64 至 SD79）上删除值 11。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X1
1	SET	F11
3	LD	X2
4	RST	F11
6	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.3.8 前沿和后沿输出（PLS, PLF）

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		常数 K, H
①			○				—

[指令符号]		[执行条件]	
PLS		PLS 指令	
PLF		PLF 指令	

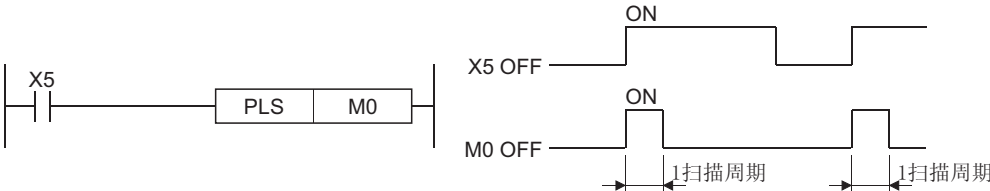
[设定数据]

设定数据	含义	数据类型
①	脉冲变换软元件	位

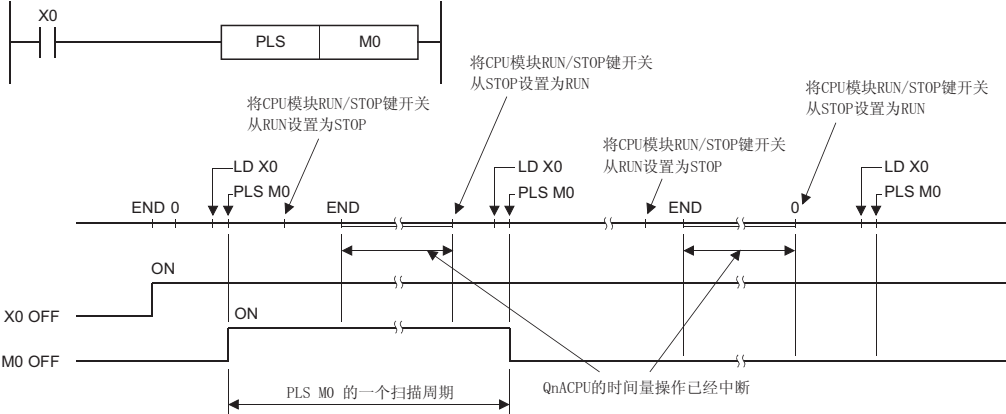
[功能]

PLS

- (1) 当从关闭转为开启时，PLS 指令将指定的软元件开启，在除从关闭到开启以外的其他情况下（如从 ON 到 ON，从 ON 到 OFF 或从 OFF 到 OFF），将指定的软元件关闭。
在一个扫描周期内，被①指定的软元件中有一个 PLS 指令，那么这个指定的软元件在扫描周期内开启。
如果在一个扫描周期内对同一个软元件执行了多于一次的 PLS 指令，这种情况执行的操作，请参见章节 3.9。



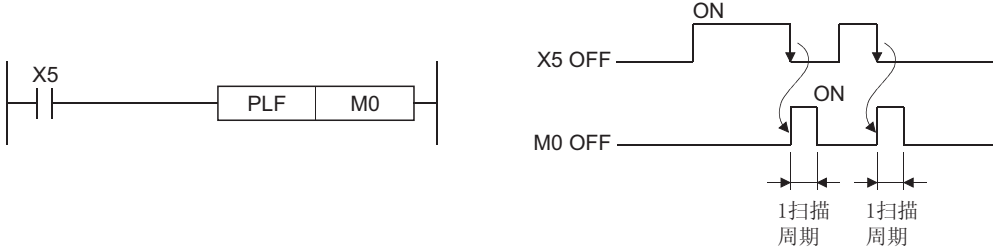
- (2) 如果在执行了 PLS 指令之后，RUN/STOP 键开关从 RUN 变换到 STOP，那么即使开关被重新设置为 RUN，PLS 指令也不会再次执行。



- (3) 当锁存继电器（L）为 PLS 指令指定后，因为在第一次扫描中 PLS 指令从关闭变为开启，开启锁存继电器来关闭电源，然后再次开启电源，执行 PLS 指令来开启指定的软元件。
在电源在下一个 PLS 指令处有开启转换到关闭后，软元件在第一个扫描周期内开启。

PLF

- (1) 当从开启变为关闭时，PLF 指令将指定的软元件开启，在除从开启到关闭以外的其他情况下（例如，从 OFF 到 OFF，从 OFF 到 ON 或从 ON 到 ON），该指令将指定的软元件关闭。
在一个扫描周期内被⑤指定的软元件有一个 PLF 指令时，在该扫描周期内，开启指定的软元件。
如果同一个软元件在一个扫描周期内 PLF 指令执行多于一次，执行操作请参见章节 3.9。



- (2) 如果在 PLF 指令执行之后，RUN/STOP 键开关从 RUN 变为 STOP，那么即使开关重新设置为 RUN，PLF 指令也不会被再次执行。

要点

请注意如果 PLS 或 PLF 指令被 CJ 指令跳过或如果在 PLS/PLF 指令执行的位置的副程序没有被 CALL 指令回调，那么被⑤指定的软元件开启时间会超过一个扫描周期。

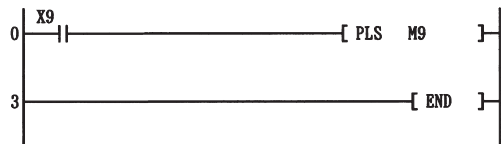
[运行错误]

- (1) 没有与 PLS 或 PLF 指令相关的操作错误。

[编程示例]

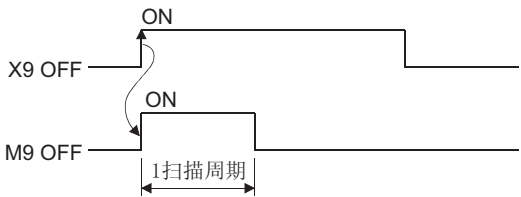
- (1) 当 X9 接通时，下列程序执行 PLS 指令。

[梯形图模式]



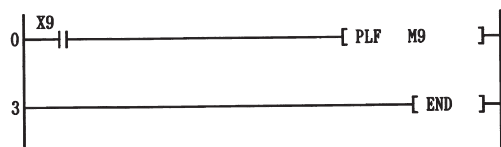
[列表模式]

步	指令	软元件
0	LD	X9
1	PLS	M9
3	END	



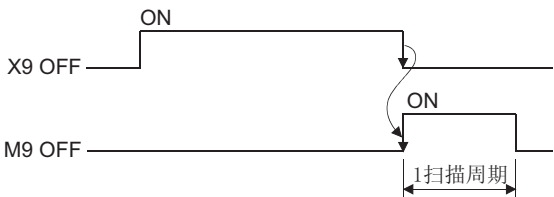
- (2) 当 X9 断开的时候，下列程序执行 PLF 指令。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X9
1	PLF	M9
3	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.3.9 位软元件输出取反（FF）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][]G[]	变址寄存器 Zn	常数 K, H	其他
	位	字		位	字				DY
⑤	○						—		○



[设定数据]

设定数据	含义	数据类型
①	待取反的软元件号	位

[功能]

(1) 当取反指令从 OFF 变为 ON 时，被①指定的软元件状态被反转

软元件	软元件状态	
	在 FF 执行之前	在 FF 执行之后
位软元件	OFF	ON
	ON	OFF
字软元件位指定	0	1
	1	0

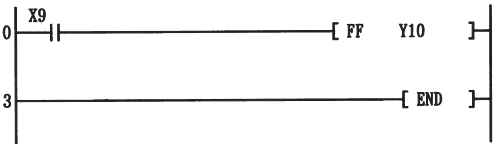
[运行错误]

(1) 没有与 FF 指令相关的运行错误。

[程序示例]

(1) 在 X9 接通时，下列程序将 Y10 的输出取反。

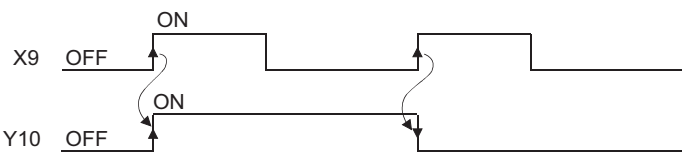
[梯形图模式]



[列表模式]

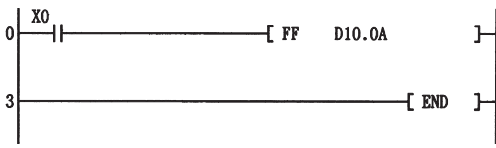
步	指令	软元件
0	LD	X9
1	FF	Y10
3	END	

[时序图]



(2) 当 X0 接通时，下列程序将 D10 的 b10（位 10）取反。

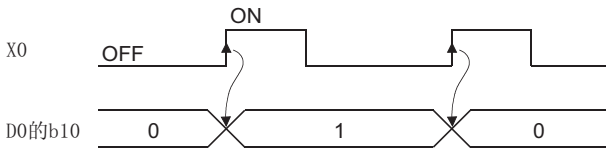
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	FF	D10.0A
3	END	

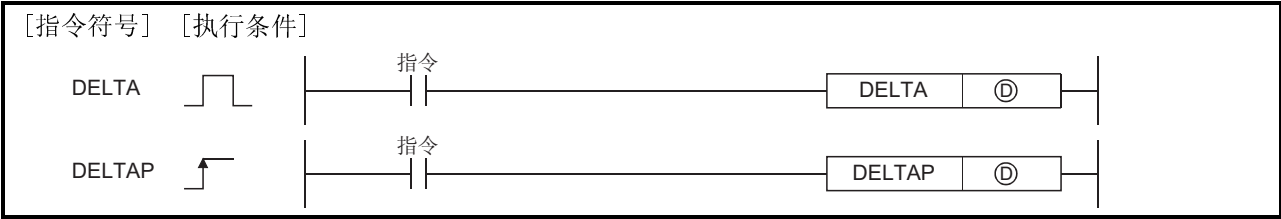
[时序图]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.3.10 直接输出的脉冲变换(DELTA, DELTAP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常数 K, H	其他
	位	字		位	字				DY
①	—								○

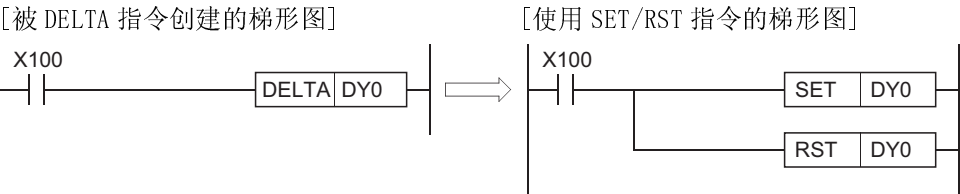


[设定数据]

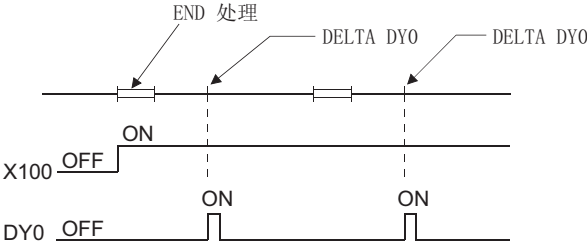
设定数据	含义	数据类型
①	将要进行脉冲变化的位	位

[功能]

- (1) 产生被①指定的直接访问输出（DY）的脉冲输出
- 如果 DELTA DY0 已经被指定，那么导致的操作将会等同于下面所示的梯形图，该梯形图使用 SET/RST 指令。



[操作]



- (2) DELTA (P) 指令是被智能模块/特殊功能模块的前沿执行使用的指令

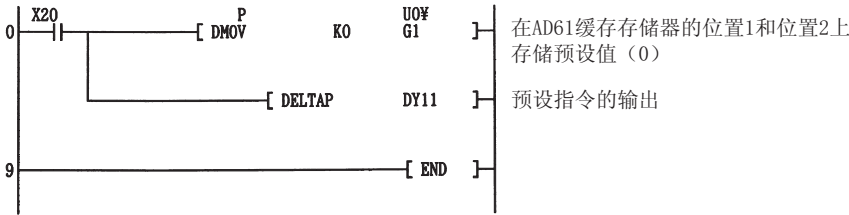
[运行错误]

- (1) 在下列情况下，会出现一个运行错误，错误标志（SM0）开启，并且错误代码存储在 SD0。
- 被Ⓓ指定的直接访问输出号超出了 CPU 模块输出范围。（错误代码：4101）

[程序示例]

- (1) 当 X20 接通的时候，下列程序预置安装在主基板的 0 插槽的 AD61 的 CH1。

[梯形图模式]



[列表模式]

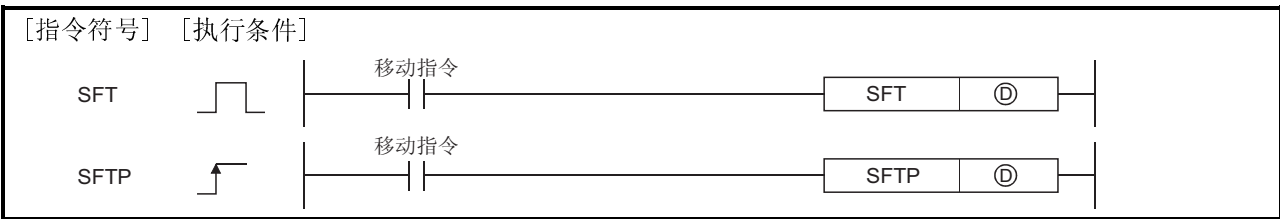
步	指令	软元件
0	LD	X20
1	DMOVP	K0
6	DELTAP	UO#G1
9	END	DY11

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.4 移动指令

5.4.1 位软元件移动（SFT，SFTP）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常数 K, H	其他
	位	字		位	字				DY
⑤	○ (除了 TC)						—		○

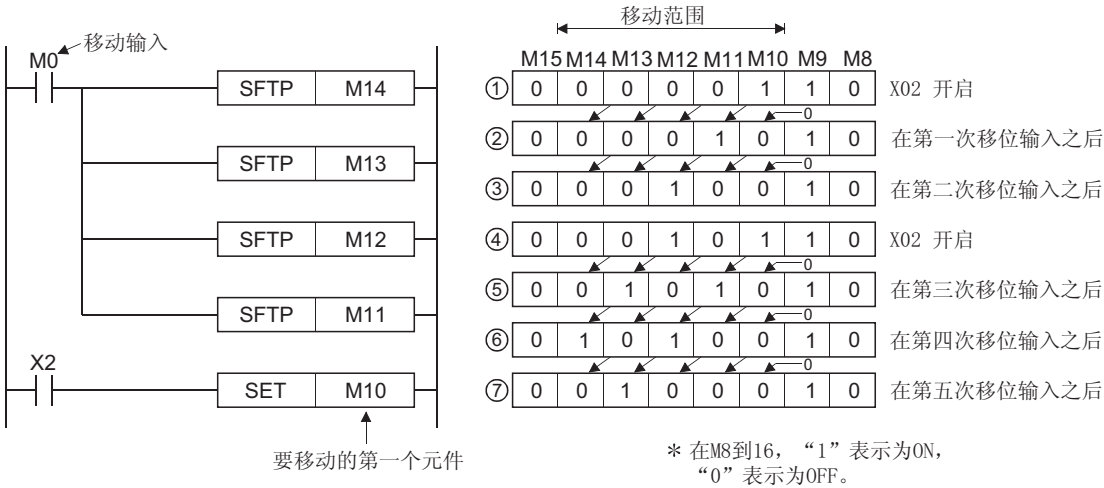


[设定数据]

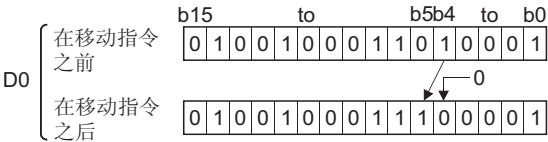
设定数据	含义	数据类型
⑤	要移动的软元件号	位

[功能]

- (1) 当使用位软元件时
- (a) 在指定的软元件之前立即将软元件的 ON/OFF 状态移动到被⑤指定的软元件中去，并且将先前的软元件关闭。
- 例如，如果 M11 已经被 SFT 指令指定，当 SET 指令被执行，那么它将会移动 M10 的 ON/OFF 状态至 M11，并且将 M10 关闭。
- (b) 使用 SET 指令将要移动的第二个软元件开启
- (c) 当 SFT 和 SFTP 被连续使用时，将从有较大号的软元件处启动程序。



- (2) 当使用字软元件的位指定
- (a) 在指定的软元件之前直接将位的 1/0 状态移动到被⑤指定的软元件的位中，并且将之前的位设置为 0。
- 例如，如果 D0.5（D0 的位 5 [b5]）已经被 SFT 指令指定，当这个 SFT 指令被执行时，将会移动 D0 的 b4 的 1/0 状态至 b5，并且将 b4 设置为 0。



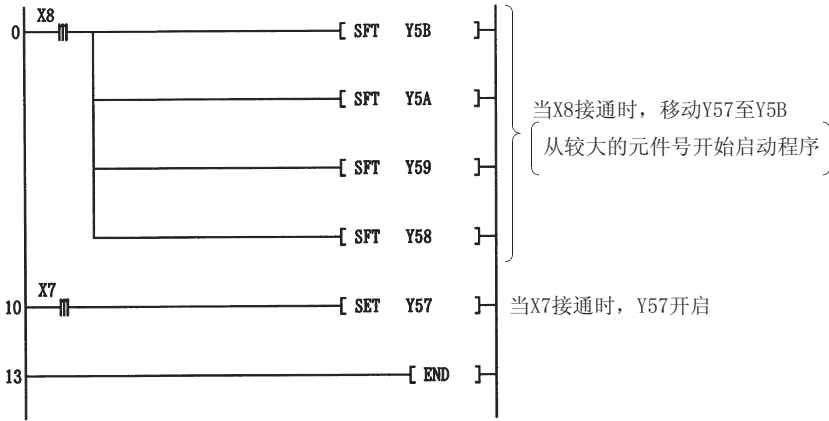
[运行错误]

- (1) 没有与 SFT(P) 指令相关的运行错误。

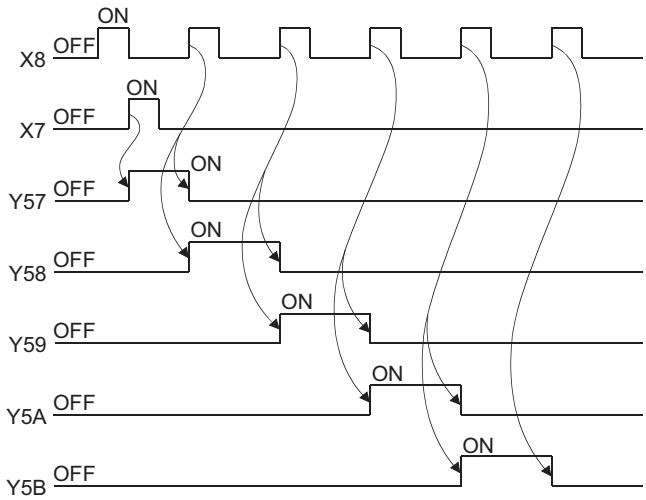
[程序示例]

- (1) 当 X8 接通的时候，下列程序将 Y57 移动到 Y5B。

[梯形图模式]



[时序图]



[列表模式]

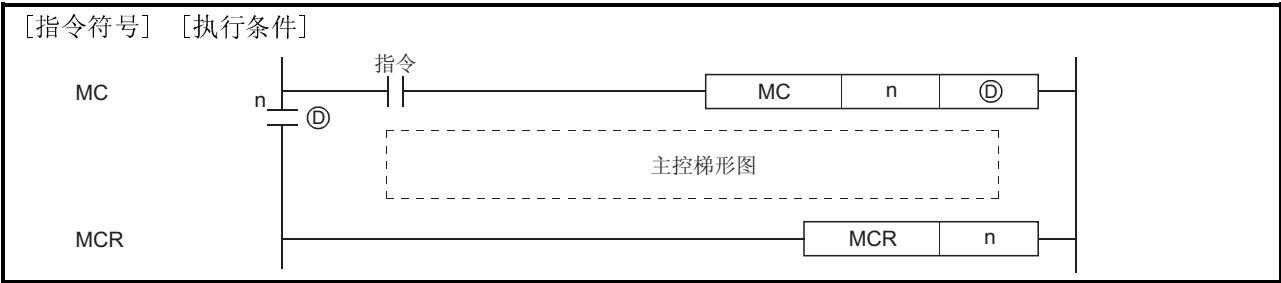
步	指令	软元件
0	LDP	X8
2	SFT	Y5B
4	SFT	Y5A
6	SFT	Y59
8	SFT	Y58
10	LDP	X7
12	SET	Y57
13	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.5 主控指令

5.5.1 设置和复位主控（MC，MCR）

设定 数据	可用软元件									
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][]G[][]	变址寄存器 Zn	常数 K, H	其他	
	位	字		位	字				N	DY
n	—						—	○	—	
⑤	○						—	—	○	

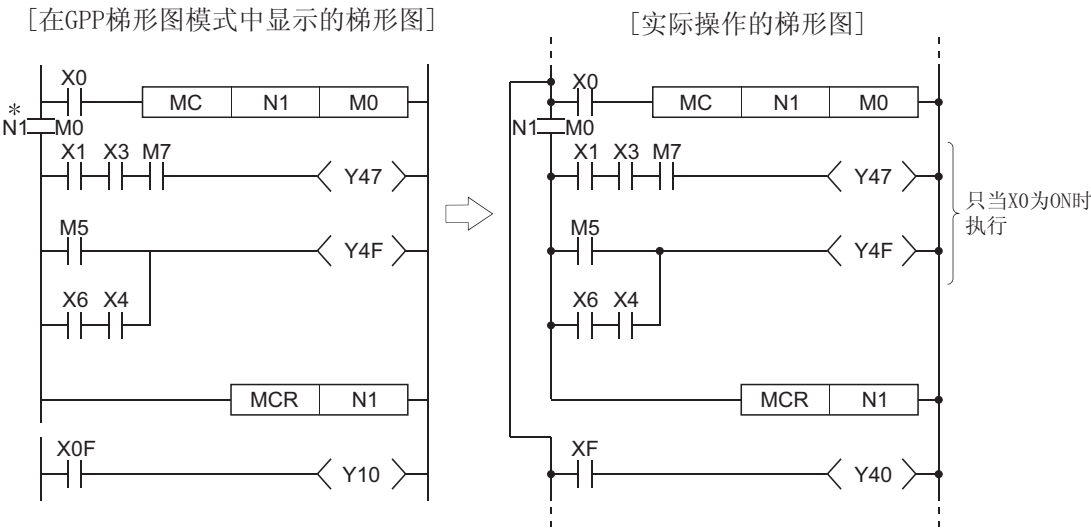


[设定数据]

设定数据	含义	数据类型
n	嵌套(N0 至 N14)	嵌套
Ⓓ	开启的软元件号	位

[功能]

通过为梯形图开启或关闭一个公共总线，主控指令被用来生成高效的梯形图顺控程序。
使用主控的梯形图，如下所示：



备注

*: 当在一个外围设备中的梯形图模式中编程时，不需要在垂直母线上输入触点。
在生成了梯形图之后，执行“变换”指令时将会自动显示，然后被设置成“读取”模式。

MC

(1) 如果当主控开始时，MC 指令的 ON/OFF 指令是 ON，那么在 MC 指令和 MCR 指令之间的运算结果将正好和所示的指令（梯形图）一样。

如果 MC ON/OFF 显示为 OFF，那么在 MC 和 MCR 指令之间的运算结果将如下所示：

软元件	软元件状态
高速定时器 低速定时器	计数器值变为 0，线圈和触点都断开。
高速累计定时器 低速累计定时器 计数器	线圈断开，但计数器的数值和触点都保持当前状态。
在 OUT 指令中的软元件	全部关闭
SET, RST SFT 基本, 应用	在下列指令中的软元件保持当前状态

(2) 当 MC 指令为 OFF 时，从 MC 指令到 MCR 指令的指令都将会被执行，所以扫描周期时间将不会缩短。

要点

如果在梯形图中有不需要的触点指令（FOR - NEXT, EI, DI, 等）使用了主控，那么不管 MC 指令的 ON/OFF 处在什么状态，CPU 模块都会执行这些指令。

(3) 通过改变被⑤指定的软元件，MC 指令能够任意次使用同一个嵌套号（N）。

(4) 当 MC 指令是 ON 时，被⑤指定的软元件线圈被开启。并且，在 OUT 指令或其他指令中使用这些相同的软元件将会使线圈变成双重线圈，所以被⑤指定的软元件不应在其他指令中使用。

MCR

(1) 这是从主控中恢复的指令，并且显示操作主控范围的结束。

(2) 在 MCR 指令之前不要有触点指令。

(3) 使用具有相同的嵌套号的 MC 指令和 MCR 指令作为配套。

然而，当 MCR 指令在一个位置嵌套时，所有的主控都能因使用最低的嵌套（N）号而终止。
（参见程序范例中的“嵌套注意事项”）

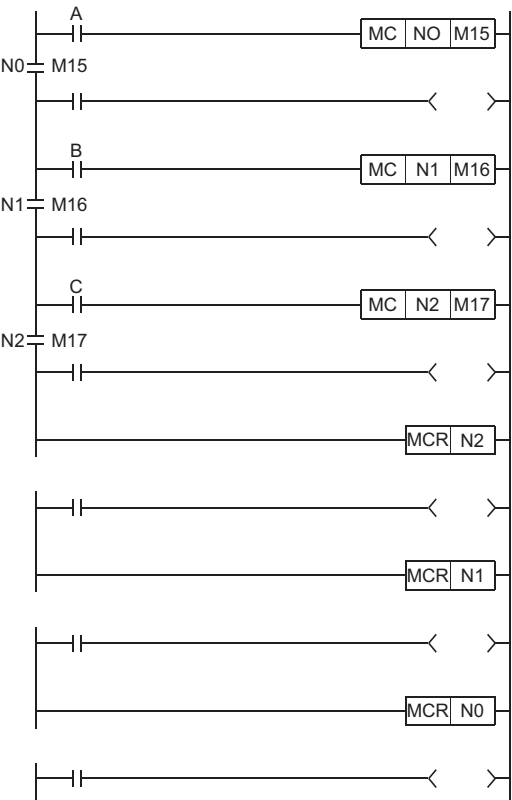
[运行错误]

(1) 没有与 MC 或 MCR 指令相关的运行错误。

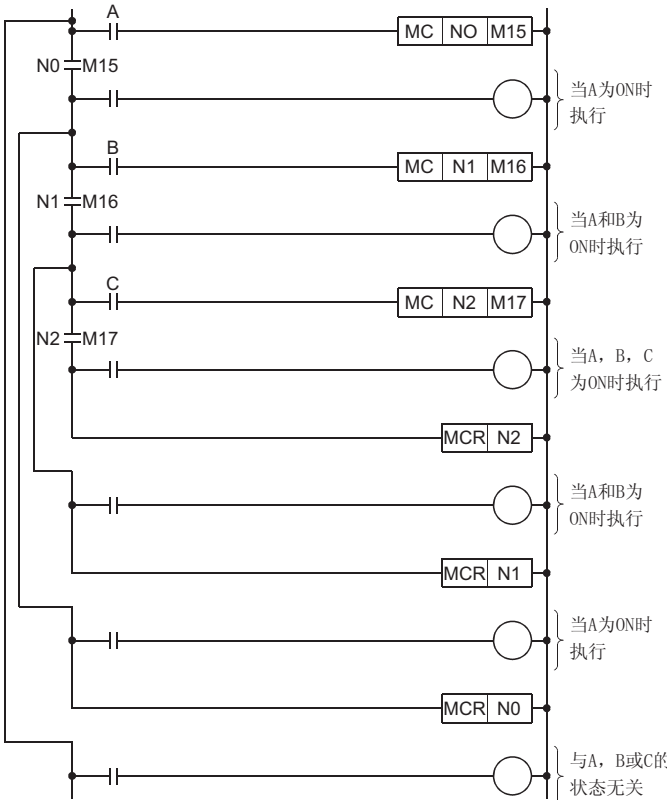
[程序示例]

主控指令能在嵌套中使用。
 嵌套 (N) 把不同的主控区域区分开来。
 嵌套能在 N0 到 N14 上执行。
 使用嵌套能够生成梯形图，这些梯形图连续地限制程序的执行条件。
 有嵌套的梯形图如下所示：

[在 GPP 梯形图模式中显示的梯形图]



[实际操作的梯形图]



使用嵌套结构的注意事项

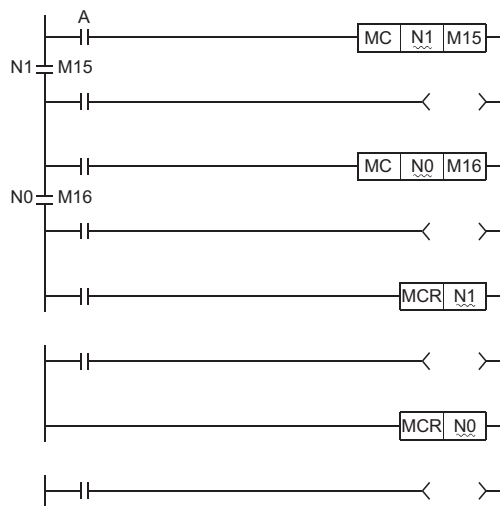
(1) 嵌套能使用达 15 次 (N0 至 N14)

使用嵌套时，在 MC 指令中嵌套是按嵌套号 (N) 从低到高插入的，在 MCR 指令中是按嵌套号从高到低插入的。

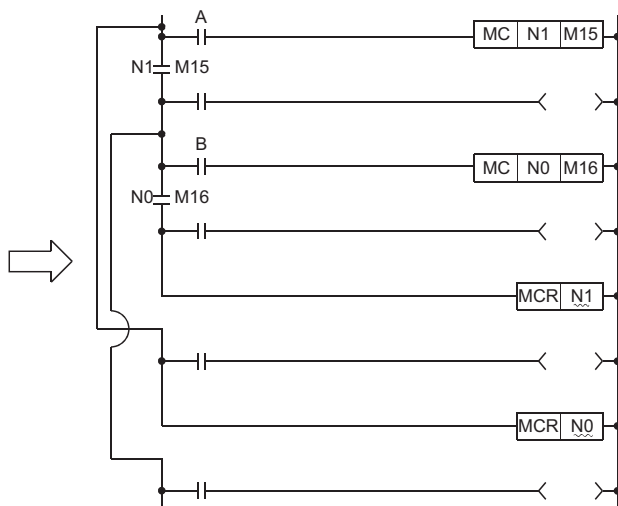
如果该顺序逆向了，将不会有嵌套结构，并且 CPU 模块也不能进行正确的操作。

例如，如果嵌套是被 MC 指令从 N1 到 N0 的顺序指定的，并且同样被 MCR 指定从 N1 到 N0 的顺序指定，那么垂直母线将会交叉并且不会产生正确的主控梯形图。

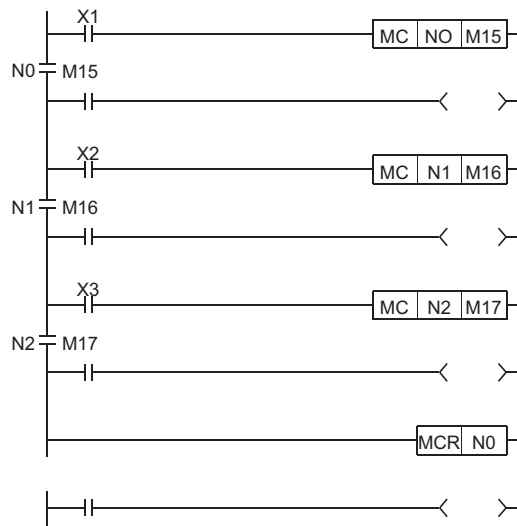
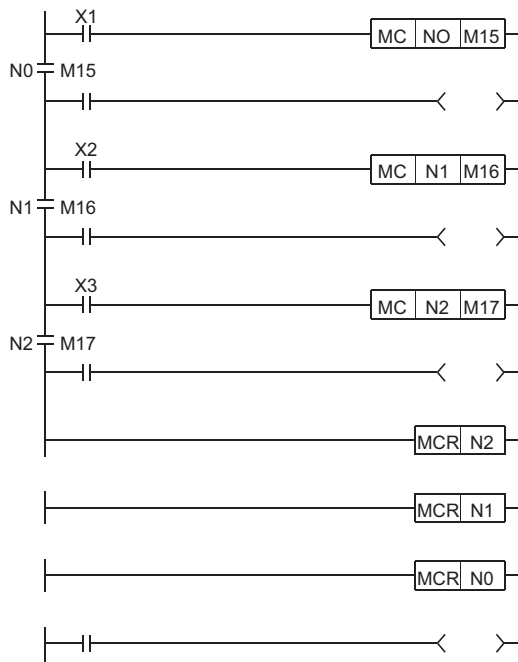
[在 GPP 梯形图模式中显示的梯形图]



[实际操作的梯形图]



(2) 如果嵌套结构导致了 MCR 指令集中在一个位置，那么所有的主控均会因使用最低的嵌套号 (N) 而被终止。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.6 结束指令

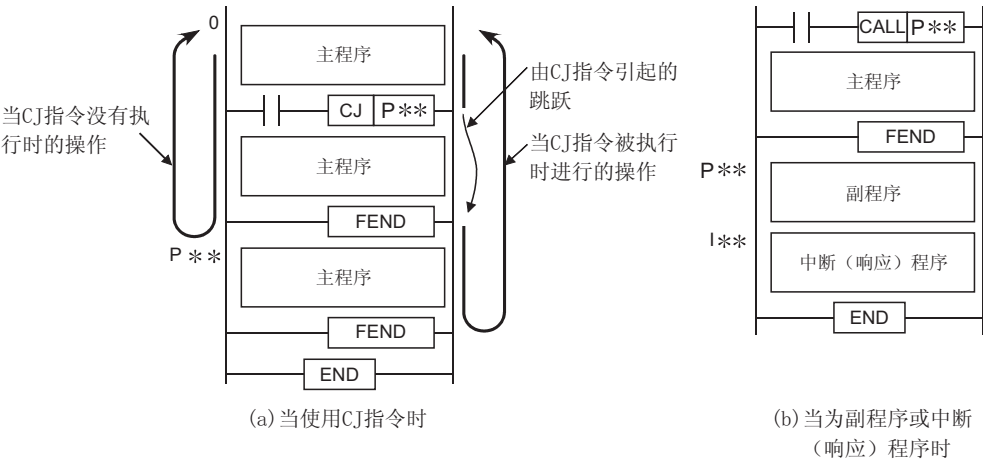
5.6.1 结束主程序（FEND）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常数	其他
	位	字		位	字				
—	—								

[指令符号]	[执行条件]
FEND	

[功能]

- (1) 当 CJ 指令或其他指令被用来在顺控程序操作中创建一个分支时，在这种情况下使用 FEND 指令。并且当主程序从副程序或暂停（响应）程序中分离出来时，在这种情况下也使用 FEND 指令
- (2) FEND 指令的执行将会使得 CPU 模块结束正在执行的程序。
- (3) 在外围设备中甚至在 FEND 指令之后的顺控程序都能够在梯形图显示中显示出来。（外围设备持续显示梯形图直到遇到 END 指令）



[运行错误]

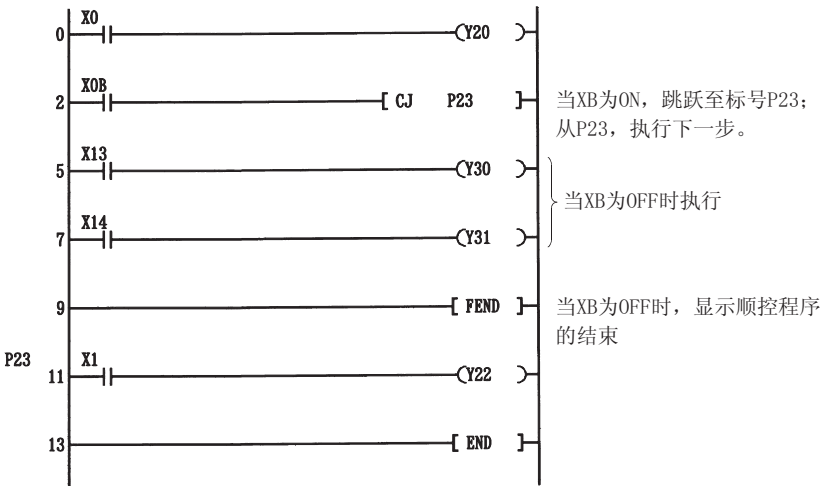
- (1) 在下列情况下会出现运行错误，错误标志（SM0）开启，并且将错误代码存储在SD0中。
 - FEND 指令在 CALL, FCALL, ECALL, 或 EFCALL 指令执行之后，并且在 RET 指令之前执行。（错误代码：4211）
 - FEND 指令在 FOR 指令执行之后且在 NEXT 指令执行之前执行。（错误代码：4200）

- FEND 指令在暂停（响应）程序中执行，且在 IRET 指令执行之前执行。（错误代码：4221）
- FEND 指令在 CHKCIR 和 CHKEND 指令之间执行。（错误代码：4230）
- FEND 指令在 IX 和 IXEND 指令之间执行。（错误代码：4231）

[程序示例]

(1) 下列程序使用 CJ 指令

[梯形图模式]



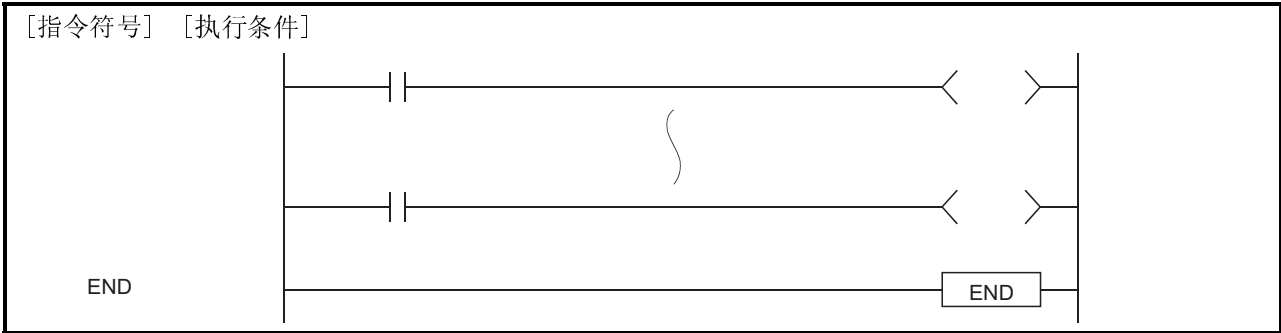
[列表模式]

步	指令	软元件
0	LD	X0
1	OUT	Y20
2	LD	X0B
3	CJ	P23
5	LD	X13
6	OUT	Y30
7	LD	X14
8	OUT	Y31
9	FEND	
10		P23
11	LD	X1
12	OUT	Y22
13	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

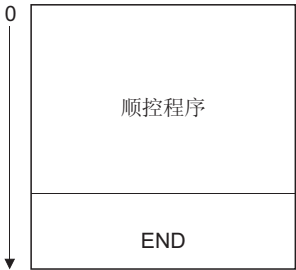
5.6.2 结束顺控程序（END）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][]	变址寄存器 Zn	常数 K, H	其他
	位	字		位	字				U
	—								



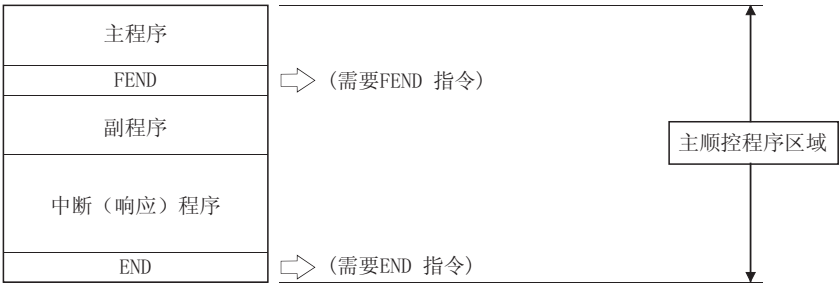
[功能]

- (1) 表示程序的结束，包括主程序，副程序和暂停（响应）程序。
END 指令的执行将会引起 CPU 模块结束正在执行的程序。



- (2) 在主顺控程序的执行过程中不能使用 END 指令。
如果在程序执行过程中需要进行 END 处理，请使用 FEND 指令。
- (3) 当在外部设备的梯形图模式中编程时，不需要输入 END 指令。

(4) 在主程序，副程序和暂停（响应）程序中，使用 END 和 FEND 指令被暂停如下：



[运行错误]

(1) 在下列情况下会出现运行错误，错误标志（SM0）开启，并且将错误代码存储在 SD0 处。

- END 指令在 RET 指令执行之前且在 CALL, FCALL, ECALL, 或 EFCALL 指令执行之后执行。(错误代码: 4211)
- END 指令在 NEXT 指令执行之前且在 FOR 指令执行之后执行。(错误代码: 4200)
- END 指令在暂停（响应）程序中执行，且先于 IRET 指令的执行。(错误代码: 4221)
- END 指令在 CHKCIR 到 CHKEND 指令循环中执行。(错误代码: 4230)
- END 指令在 IX 到 IXEND 指令循环中执行。(错误代码: 4231)

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.7 其他指令

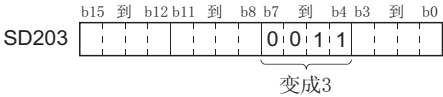
5.7.1 顺控程序暂停(STOP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][6][]	变址寄存器 Zn	常数 K, H	其他
	位	字		位	字				U
	—								



[功能]

- (1) 当暂停输入被开启，输出 Y 被复位并且 CPU 模块的操作被停止。
(如果 RUN/STOP（键）开关被转为 STOP 设置那么会出现相同的结果。)
- (2) 执行 STOP 指令将会使特殊寄存器 SD203 b4 至 b7 的值变为 3。



- (3) 为了能在执行 STOP 指令之后重新执行 CPU 模块操作，应复位已经从 RUN 改变到 STOP 的 RUN/STOP 键开关，使其重新为 RUN。

[运行错误]

- (1) 在下列情况下会出现运行错误，错误标志（SM0）开启，并且错误代码储存至 SD0。
- STOP 指令在 RET 指令执行之前且在 CALL/FCALL/ECALL/EFCALL/XCALL 指令执行之后执行。

(错误代码: 4211)

● STOP 指令在 NEXT 指令执行之前且在 FOR 指令执行之后执行。

(错误代码: 4200)

● STOP 指令在暂停（响应）程序中并且在 IRET 指令执行之前执行。

(错误代码: 4221)

● STOP 指令在 CHKCIR 到 CHKEND 指令循环中执行。

(错误代码: 4230)

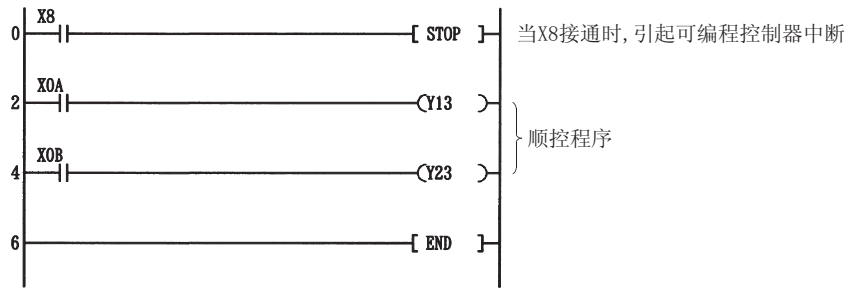
● STOP 指令在 IX 到 IXEND 指令循环中执行。

(错误代码: 4231)

[程序示例]

(1) 当 X8 接通时，下列程序将 CPU 模块停止。

[梯形图模式]



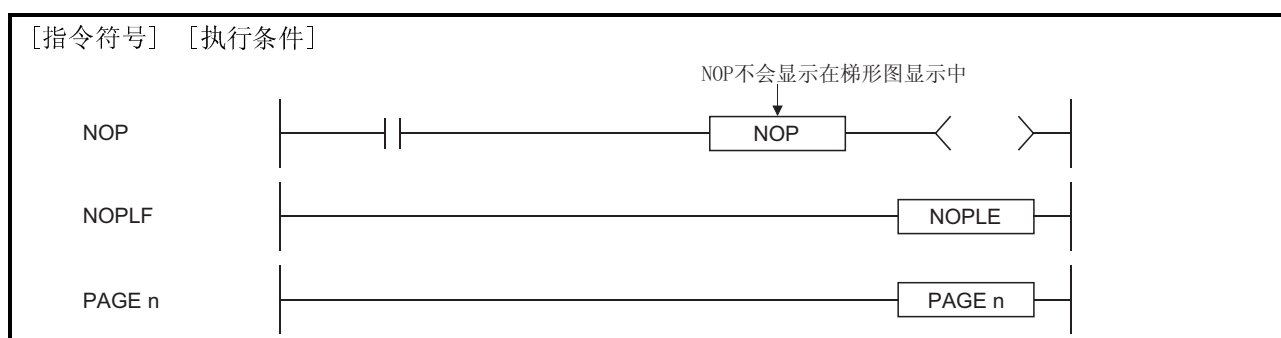
[列表模式]

步	指令	软元件
0	LD	X8
1	STOP	
2	LD	X0A
3	OUT	Y13
4	LD	X0B
5	OUT	Y23
6	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

5.7.2 无处理(NOP, NOPLF, PAGE n)

设定 数据	可用软元件								
	内部软元件 (系统，用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]N[]		特殊功能模 块 U[]G[]	变址寄存器 Zn	常数 K, H	其他
	位	字		位	字				
—	—								



[功能]

NOP

- (1) 这是一个无处理指令，到该点之前不会对任何处理有影响。
- (2) NOP 指令适用于下列情况：
 - (a) 为顺控程序调试插入空间。
 - (b) 删除一个指令，但是不用改变步号（用 NOP 替换指令）
 - (c) 临时性删除一个指令。

NOPLF

- (1) 这是一个无处理指令，到该点之前不会对任何处理有影响。
- (2) 使用 NOPLF 指令可使从外围设备中打印的页在任何需要位置改变。
 - (a) 当打印梯形图
 - 如果有 NOPLF 指令，那么在梯形图块之间将会插入一个页面暂停。
 - 如果 NOPLF 指令插在梯形图中间，那么梯形图将不能正确显示。
不要在梯形图的中间插入 NOPLF 指令。
 - (b) 当打印指令列表时
 - 在打印 NOPLF 指令之后将会被换页。
- (3) 关于外围设备的打印输出的详细资料，参见使用中的外围设备的操作手册。

PAGE n

- (1) 这是一个无处理指令，到该点之前不会对任何处理有影响。
- (2) 在 PAGE n 指令之后的程序中指定第 n 页的 0 步开始进行处理（外围设备显示，打印机等）。
- (3) 如果没有 PAGE n 指令，那么处理从第 0 页开始。

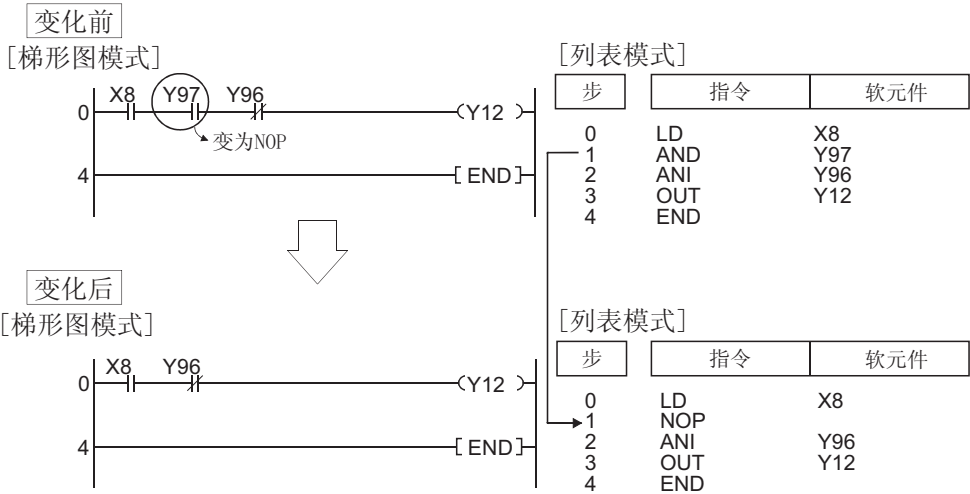
[运行错误]

- (1) 没有与 NOP，NOPLF，或 PAGE 指令相关的错误。

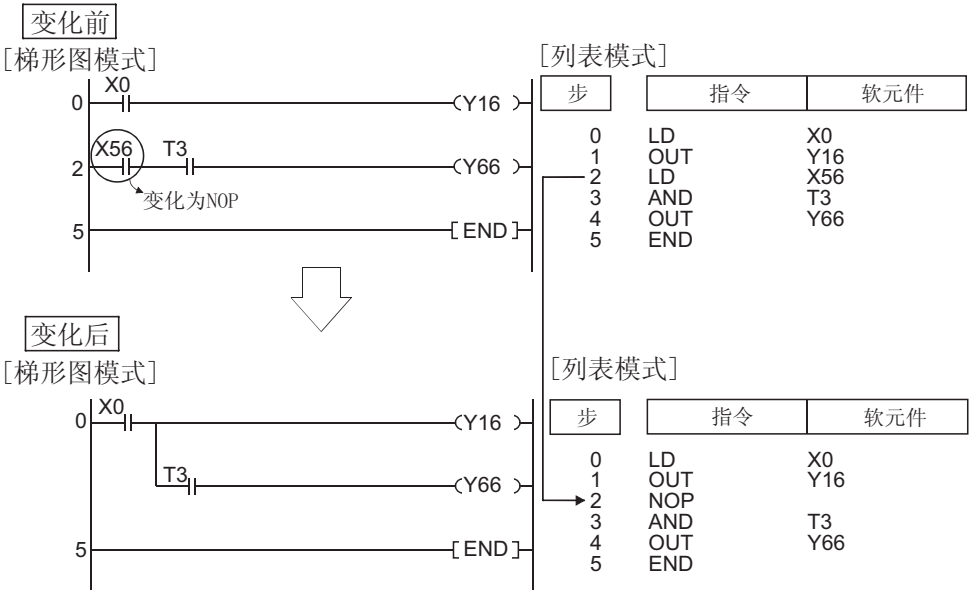
[程序示例]

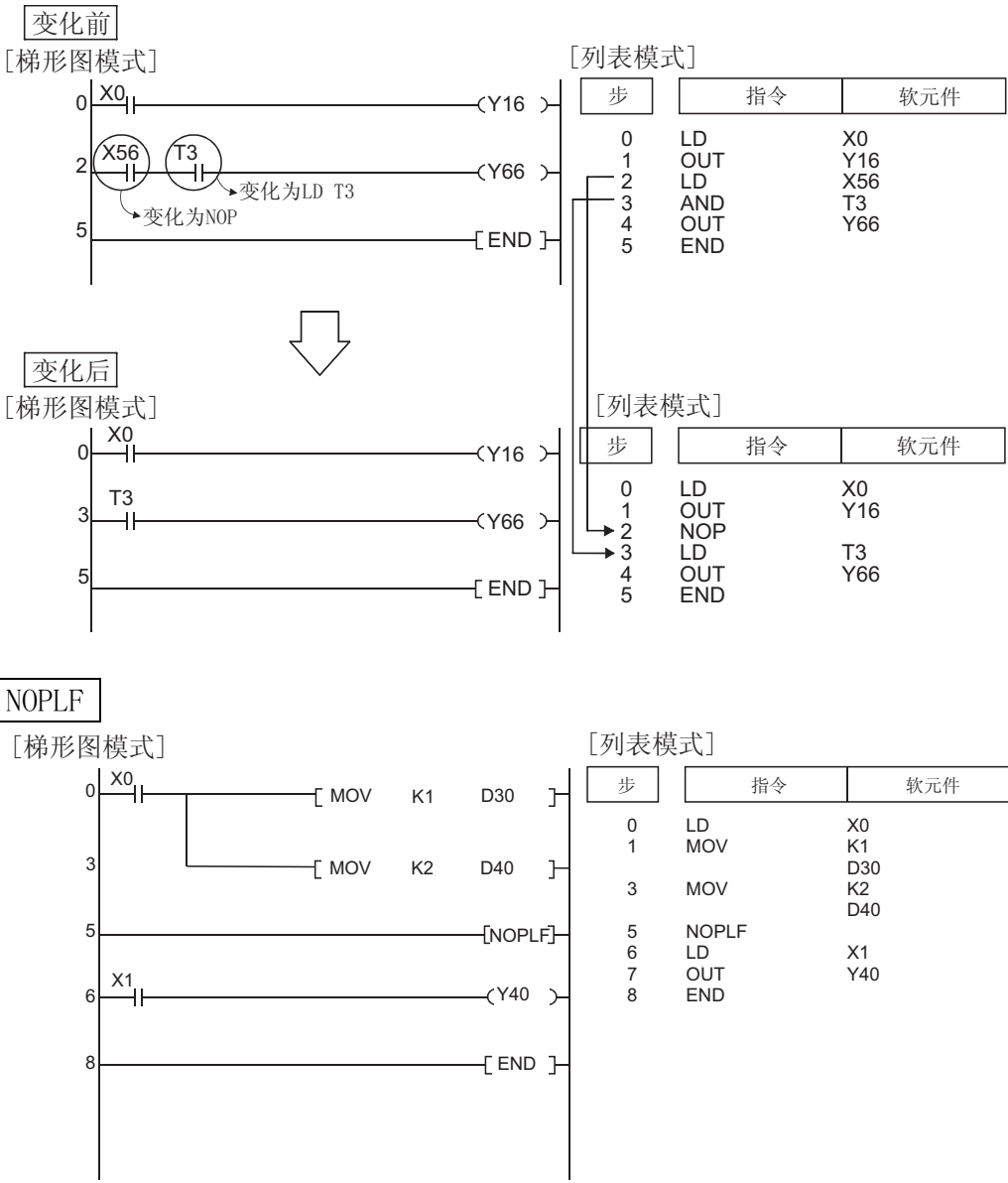
NOP

- (1) 触点关闭 删除 AND 或 ANI 指令

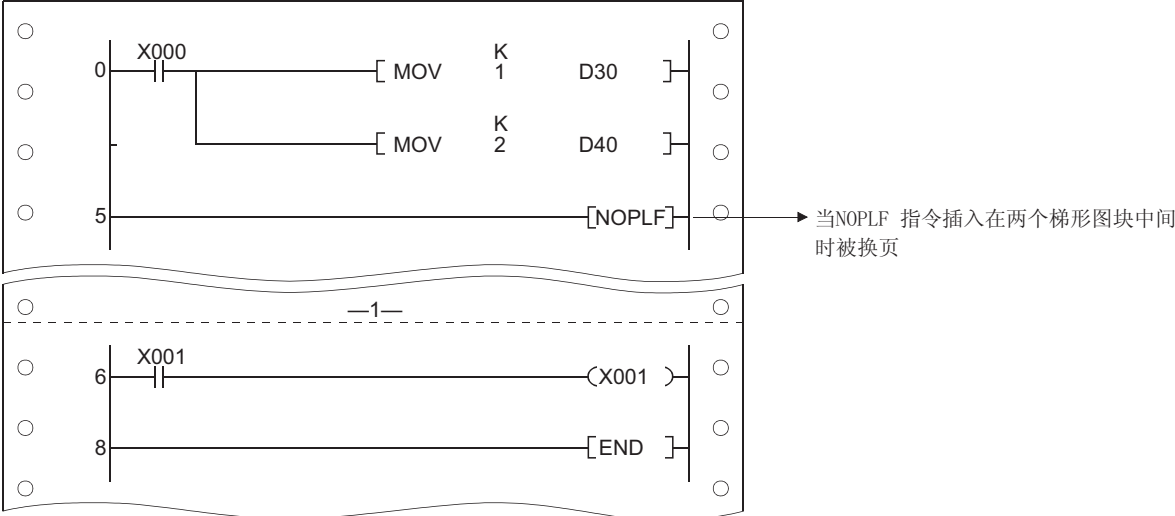


- (2) 触点关闭 LD，LDI 变换至 NOP（请注意将 LD 和 LDI 完全改变为 NOP，改变了梯形图的性质）。

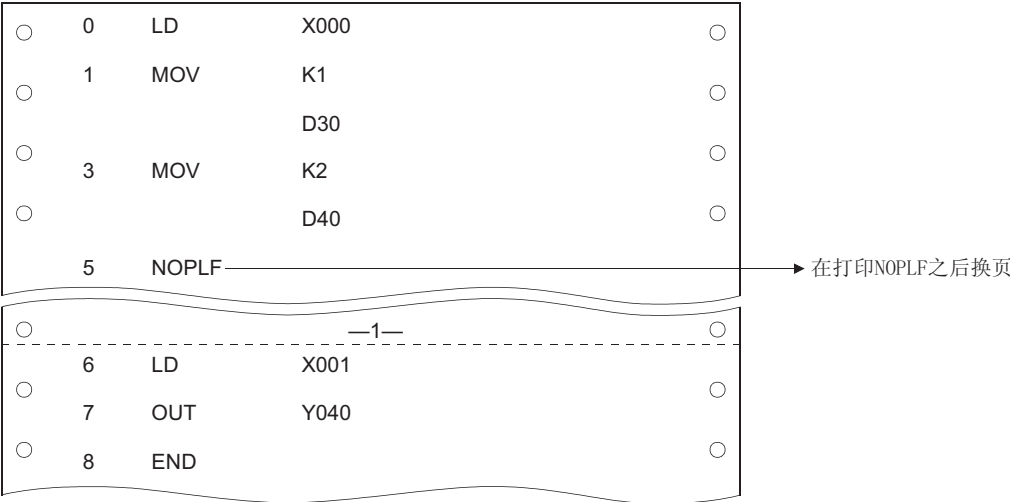




● 打印梯形图将导致以下结果：

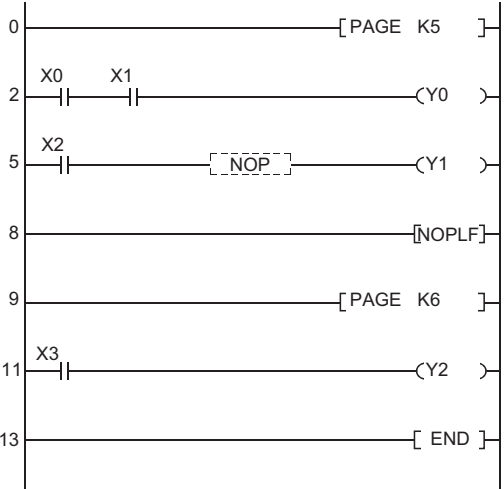


● 用 NOPLF 指令打印指令列表将会有如下结果：



PAGE n

[梯形图模式]



[列表模式]

步	指令	软元件
0	PAGE	K5
2	LD	X0
3	AND	X1
4	OUT	Y0
5	LD	X2
6	NOP	
7	OUT	Y1
8	NOPLF	
9	PAGE	K6
11	LD	X3
12	OUT	Y2
13	END	

备忘录

[illegible]

6. 基本指令

下述类型的基本指令是可以使用的。

指令	含义	参考章节
比较操作指令	数据与数据比较	6.1 节
算术运算指令	与其它数据进行加、减、乘、除、递增或递减	6.2 节
数据转换指令	转换数据类型	6.3 节
数据传送指令	传送指定的数据	6.4 节
程序分支指令	程序跳转	6.5 节
程序执行控制指令	允许和禁止程序中断	6.6 节
刷新指令	刷新位软元件	6.7 节

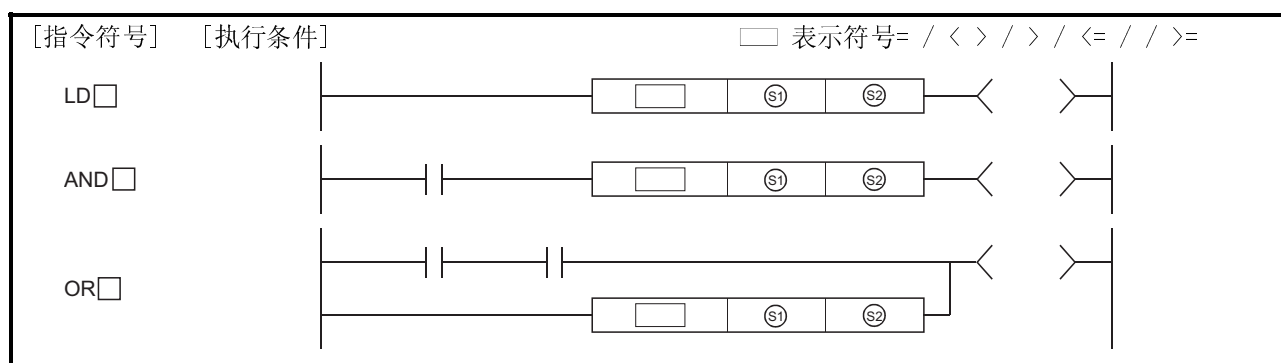
QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

*:序列号前五位是04122或更后的。

6.1 比较操作指令

6.1.1 BIN 16 位数据比较(=, <, >, <=, <, >=)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
⑧1	○								—
⑧2	○								—



[设定数据]

设定数据	含义	数据类型
⑧1	被比较的数据，或存储被比较数据的软元件号	BIN 16 位
⑧2		

「功能」

- (1) 把由 S2 指定的软元件的 BIN16 位数据和由 S1 指定的软元件的 BIN 16 位数据作为 A 触点，并进行比较运算。

- (2) 单个指令比较操作的结果如下:

指令符号 <input type="checkbox"/>	条件	比较操作结果	指令符号 <input type="checkbox"/>	条件	比较操作结果
=	$S1 = S2$	导通	\neq	$S1 \neq S2$	不导通
< >	$S1 \neq S2$		< >	$S1 = S2$	
>	$S1 > S2$		>	$S1 \leq S2$	
< =	$S1 \leq S2$		< =	$S1 > S2$	
<	$S1 < S2$		<	$S1 \geq S2$	
> =	$S1 \geq S2$		> =	$S1 < S2$	

- (3) 当十六进制常数被Ⓔ1和Ⓔ2指定时, 或最高位(b15)为 1 的数字值(8 到 F)被指定时, 此值将当作负的 BIN 值被读取, 用于比较。

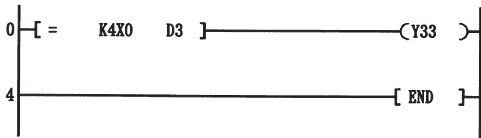
[运行错误]

(1) 没有与=, <, >, <=, >, 或 >=指令相关的运行错误。

[程序示例]

(1) 下列程序将从 X0 到 XF 的数据与 D3 的数据进行比较, 如果数据相同则将 Y33 变为 ON。

[梯形图模式]

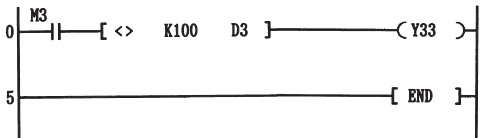


[列表模式]

步	指令	软元件
0	LD=	K4X0
3	OUT	D3
4	END	Y33

(2) 下列程序将 BIN 值 K100 与 D3 中的数据进行比较, 如果 D3 中的数据不是 100 则导通。

[梯形图模式]

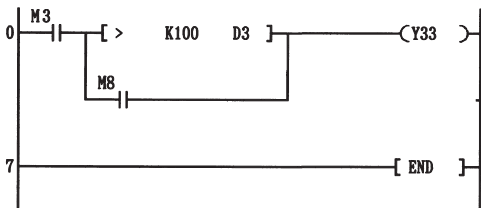


[列表模式]

步	指令	软元件
0	LD	M3
1	AND<>	K100
4	OUT	D3
5	END	Y33

(3) 下列程序将 BIN 值 100 与从 X0 到 XF 里的数据进行比较, 如果 D3 的数据小于 100 则导通。

[梯形图模式]

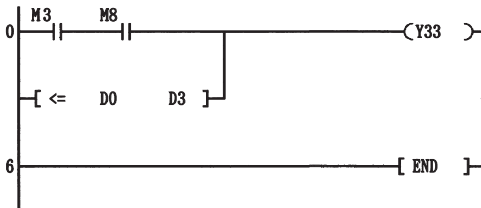


[列表模式]

步	指令	软元件
0	LD	M3
1	LD>	K100
4	OR	D3
5	ANB	M8
6	OUT	Y33
7	END	

(4) 下列程序将 D0 和 D3 里数据进行比较, 如果 D0 里的数据等于或小于 D3 里的数据则导通。

[梯形图模式]



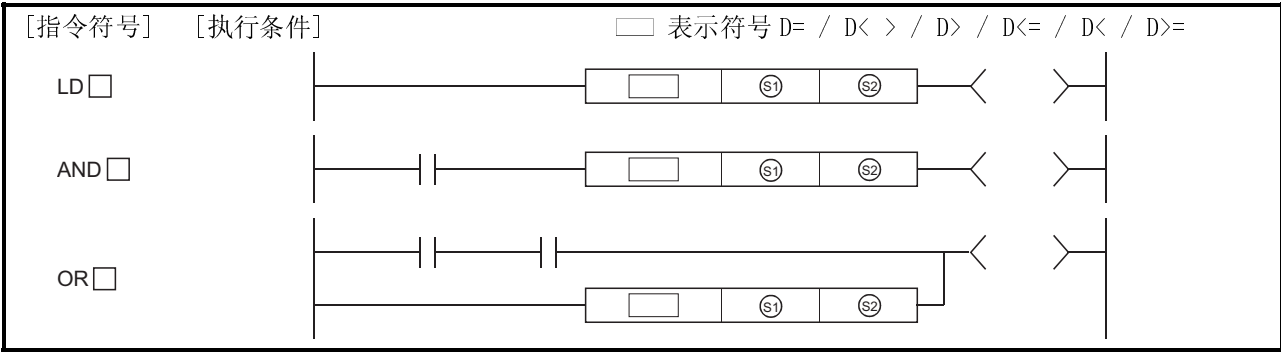
[列表模式]

步	指令	软元件
0	LD	M3
1	AND	M8
2	OR<=	D0
5	OUT	D3
6	END	Y33

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.1.2 BIN 32 位数据比较(D=, D< >, D>, D<=, D<, D>=)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
①	○								—
②	○								—



[设定数据]

设定数据	含义	数据类型
①	被比较的数据, 或存储被比较数据的软元件号	BIN 32 位
②		

[功能]

(1) 把由①指定的软元件的 BIN32 位数据和由②指定的软元件的 BIN32 位数据作为 A 触点, 进行比较操作。

(2) 单个指令比较操作的结果如下:

指令符号 □	条件	比较操作结果	指令符号 □	条件	比较操作结果
D =	① = ②	导通	D =	① ≠ ②	不导通
D < >	① ≠ ②		D < >	① = ②	
D >	① > ②		D >	① ≤ ②	
D < =	① ≤ ②		D < =	① > ②	
D <	① < ②		D <	① ≥ ②	
D > =	① ≥ ②		D > =	① < ②	

(3) 当十六进制常数被①和②指定时, 或最高位(b31)为 1 的数字值(8 到 F)被指定时, 此值将当作负的 BIN 值号被读取, 用于比较。

(4) 用于比较的数据应当由 32 位指令(DMOV 指令, 等)指定。
如果由 16 位指令(MOV 指令等)指定, 则大值和小值的比较将不能正确执行。

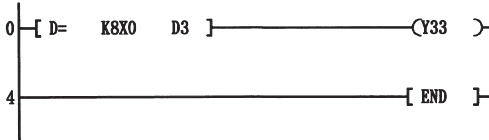
[运行错误]

(1) 没有与=, <, >, <=, <, 或 >=指令相关的运行错误。

[程序示例]

(1) 下列程序将从 X0 到 XF 的数据与 D3 中的数据进行比较, 如果数据相同则将 Y33 变为 ON。

[梯形图模式]

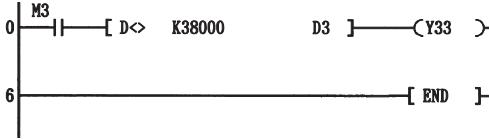


[列表模式]

步	指令	软元件
0	LDD=	K8X0 D3
3	OUT	Y33
4	END	

(2) 下列程序将 BIN 值 K38000 与 D3 和 D4 里的数据进行比较, 如果 D3 和 D4 里的数据不是 38000 则导通。

[梯形图模式]

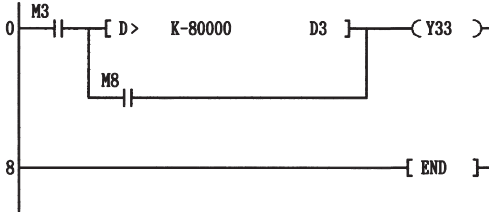


[列表模式]

步	指令	软元件
0	LD	M3
1	ANDD<>	K38000 D3
5	OUT	Y33
6	END	

(3) 下列程序将 BIN 值 K-80000 与 D3 和 D4 里的数据进行比较, 如果 D3 和 D4 里的数据小于-80000 则导通。

[梯形图模式]

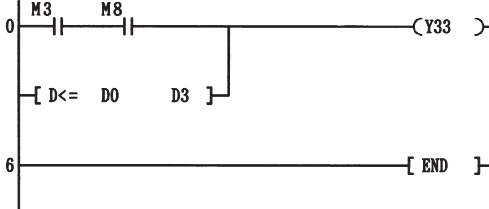


[列表模式]

步	指令	软元件
0	LD	M3
1	LDD>	K-80000 D3
5	OR	M8
6	ANB	
7	OUT	Y33
8	END	

(4) 下列程序将 D0 和 D1 里的数据与 D3 和 D4 里的数据进行比较, 如果 D0 和 D1 里的数据等于或小于 D3 和 D4 里的数据则导通。

[梯形图模式]



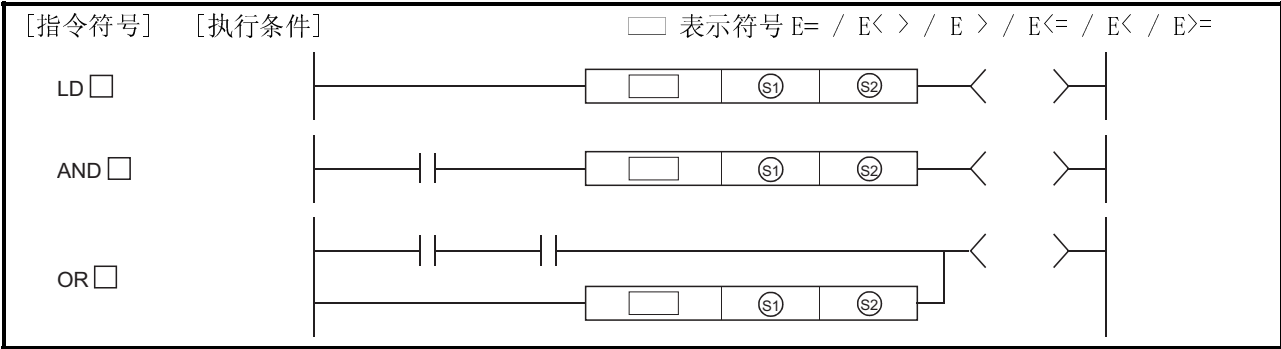
[列表模式]

步	指令	软元件
0	LD	M3
1	AND	M8
2	ORD<=	D0 D3
5	OUT	Y33
6	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

6.1.3 浮动小数点数据比较 (E=, E< >, E>, E<=, E<, E>=)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\G[]		特殊功能模 式 U[]\G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
①	—	○		—	○		—	○	—
②	—	○		—	○		—	○	—



[设定数据]

设定数据	含义	数据类型
①	被比较的数据, 或存储被比较数据的软元件号	实数
②		

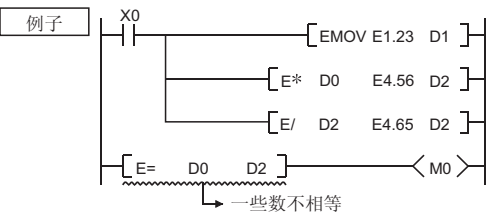
[功能]

- (1) 由①和②指定的软元件中的浮动小数点数据作为 A 触点, 进行比较操作。
- (2) 单个指令的比较操作结果如下:

□中的指令符号	条件	比较操作结果	□中的指令符号	条件	比较操作结果
E =	① = ②	导通	E =	① ≠ ②	不导通
E < >	① ≠ ②		E < >	① = ②	
E >	① > ②		E >	① ≤ ②	
E < =	① ≤ ②		E < =	① > ②	
E <	① < ②		E <	① ≥ ②	
E > =	① ≥ ②		E > =	① < ②	

要点

注意, 使用=指令时, 有时会导致由错误而引起的两个值不相等的情况。



[运行错误]

- (1) 在发生下列运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储到 SD0 中。
指定软元件的值为-0。* (错误代码：4100)

[程序示例]

- (1) 下列程序对 D0 和 D1 的浮动小数点实数与 D3 和 D4 的浮动小数点实数数据进行比较。

[梯形图模式]		[列表模式]	
		步	指令
		步	软元件
		0	LDE=
			D0
		3	OUT
		4	END
			D3
			Y33

- (2) 下列程序对浮动小数点实数 1.23 与 D3 和 D4 的浮动小数点实数数据进行比较。

[梯形图模式]		[列表模式]	
		步	指令
		步	软元件
		0	LD
		1	ANDE<>
			M3
			E1.23
		5	OUT
		6	END
			D3
			Y33

- (3) 下列程序对 D0 和 D1 的浮动小数点实数与 D3 和 D4 的浮动小数点实数数据进行比较。

[梯形图模式]		[列表模式]	
		步	指令
		步	软元件
		0	LD
		1	LDE>
			M3
			D0
			D3
		4	OR
		5	ANB
		6	OUT
		7	END
			M8
			Y3

- (4) 下列程序对 D0 和 D1 的浮动小数点实数与浮动小数点实数数据 1.23 进行比较。

[梯形图模式]		[列表模式]	
		步	指令
		步	软元件
		0	LD
		1	AND
		2	ORE<=
			M3
			M8
			D0
			E1.23
		6	OUT
		7	END
			Y33

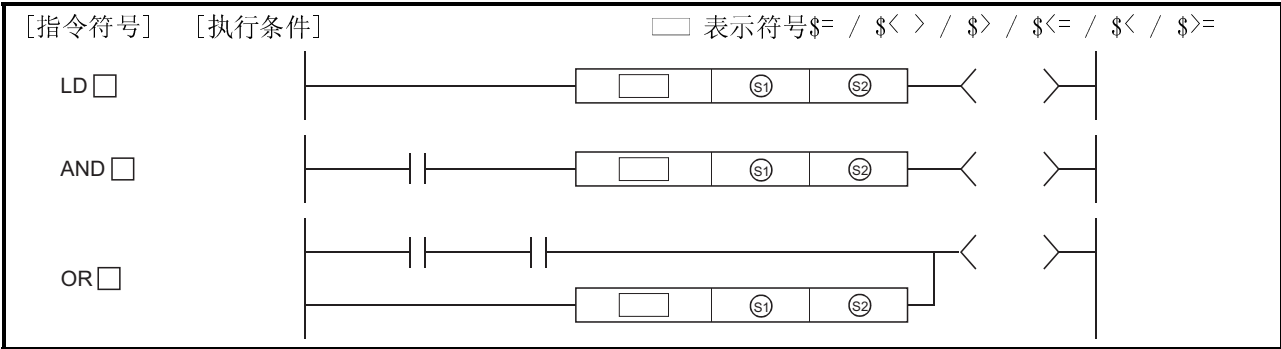
备注

*: 如果-0 被指定，则有不能导致运行错误的 CPU 模块。
详细内容参考 3.2.4 节。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

6.1.4 字符串数据比较(\$=, \$< >, \$>, \$<=, \$<, \$>=)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\[]		变址寄存器 Zn	常数 \$
	位	字		位	字		
①	—	○		—			○
②	—	○		—			○



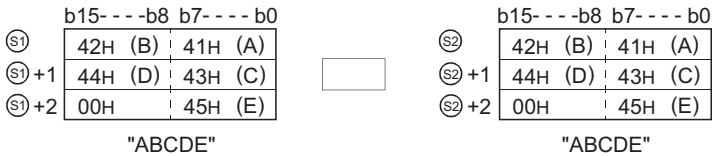
[设定数据]

设定数据	含义	数据类型
①	比较数据或存储比较数据的起始软元件号	字符串
②		

[功能]

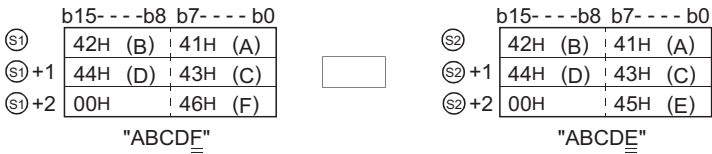
- (1) 将存储在由②指定的软元件里的字符串数据和存储在由①指定的软元件里的字符串数据作为 A 触点，并进行比较操作。
- (2) 比较操作涉及到字符串里第一个字符的 ASCII 码的字符与字符比较。
- (3) ①和②字符串包含了从指定的软元件号到下一个存储了代码“00H”的软元件号中的所有字符。

(a) 如果所有字符串都匹配，则比较结果匹配。



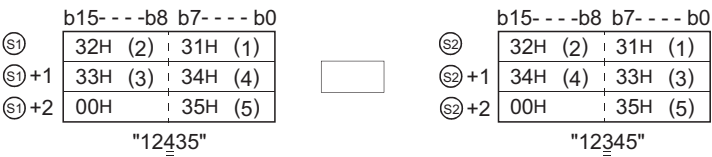
□中的指令符号	比较操作结果	□中的指令符号	比较操作结果
\$ =	导通	\$< =	导通
\$< >	不导通	\$<	不导通
\$>	不导通	\$> =	导通

(b) 如果字符串不同，则带较大字符代码的字符串将更大。



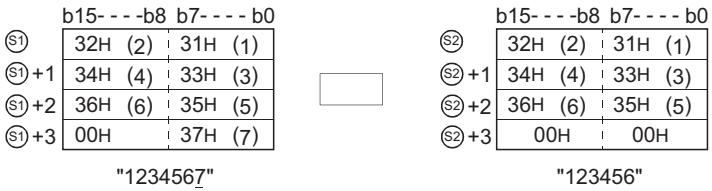
<div>中的指令符号</div>	比较操作结果	<div>中的指令符号</div>	比较操作结果
\$ =	不导通	\$< =	不导通
\$< >	导通	\$<	不导通
\$>	导通	\$> =	导通

(c) 如果字符串不同，则第一个大小不同的字符代码将决定字符串是较大还是较小。



<div>中的指令符号</div>	比较操作结果	<div>中的指令符号</div>	比较操作结果
\$ =	不导通	\$< =	不导通
\$< >	导通	\$<	不导通
\$>	导通	\$> =	导通

(4) 如果由Ⓢ1和Ⓢ2指定的字符串长度不同，则有较长字符串的数据将更大。



<div>中的指令符号</div>	比较操作结果	<div>中的指令符号</div>	比较操作结果
\$ =	不导通	\$< =	不导通
\$< >	导通	\$<	不导通
\$>	导通	\$> =	导通

[运行错误]

- (1) 在发生下列运行错误的情况中，错误标志 (SM0) 变为 ON，并且在 SD0 中存储错误代码。
- 代码 “00H” 或非匹配在由 (S1)或(S2)指定的软元件号之后的相关软元件范围内不存在。
(错误代码:4101)

要点

在进行字符串比较的同时，字符串数据比较指令也检查软元件范围。

因此，即使字符串长度超过软元件范围，字符串数据仍然被比较。这时如果在软元件范围内检测到字符不匹配，则比较操作结果不返回运行错误就输出。

例子

(S1) 数据

D12287	"B"	"A"
W0	00H	"C"

(S2) 数据

D10	"Z"	"A"
D11	00H	"C"

* 在上述例子中，(S1)字符串超过软元件范围，但因为它的第二个字符与 (S2) 中的不同，因此比较结果是(S1)不等于(S2)，并且运行结果是不导通。

在这种情况下，因为是对 D12287 (软元件范围内) 进行不导通检测，因此没有运行错误返回。

[程序示例]

- (1) 下述程序对存储在 D0 之后的字符串和 D10 之后的字符进行比较。

[梯形图模式]

[列表模式]

步	指令	软元件
0	LD\$=	D0
		D10
3	OUT	Y33
4	END	

- (2) 下述程序对字符串 “ABCDEF” 和存储在 D10 之后的字符串进行比较。

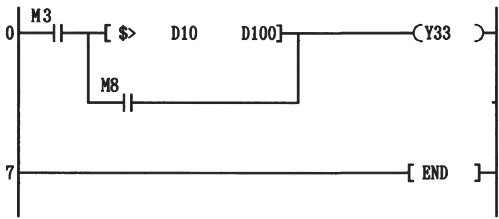
[梯形图模式]

[列表模式]

步	指令	软元件
0	LD	M3
1	AND\$<>	"ABCDEF"
		D10
7	OUT	Y33
8	END	

(3) 下述程序对存储在 D10 之后的字符串和存储在 D100 之后的字符串进行比较。

[梯形图模式]

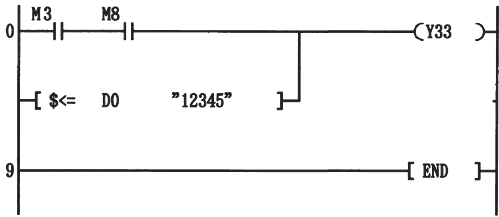


[列表模式]

步	指令	软元件
0	LD	M3
1	LD\$>	D10 D100
4	OR	M8
5	ANB	
6	OUT	Y33
7	END	

(4) 下述程序对存储在 D10 之后的字符串和字符串 “12345” 进行比较。

[梯形图模式]




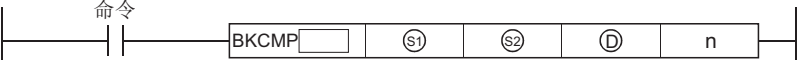

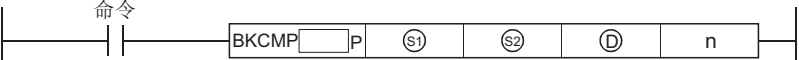
[列表模式]

步	指令	软元件
0	LD	M3
1	AND	M8
2	OR\$<=	D0 "12345"
8	OUT	Y33
9	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.1.5 BIN 块数据比较 (BKCMP, BKCMP□P)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J□□□		特殊功能模 块 U□□□G□□	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
①	—	○			—			○	—
②	—	○			—			—	—
③	○	○			—			—	—
n	○	○			○			○	—

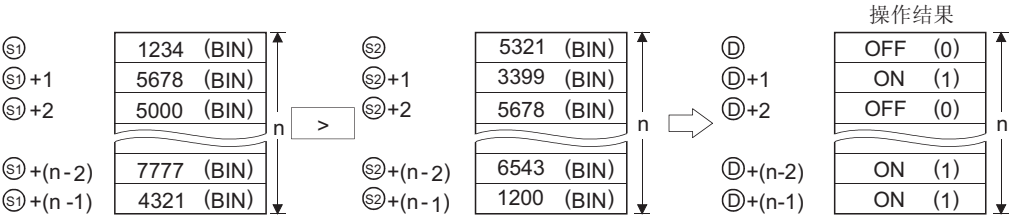
[指令符号] [执行条件]		□ 表示符号= / < > / > / <= / < / >=	
BKCMP□			
BKCMP□P			

[设定数据]

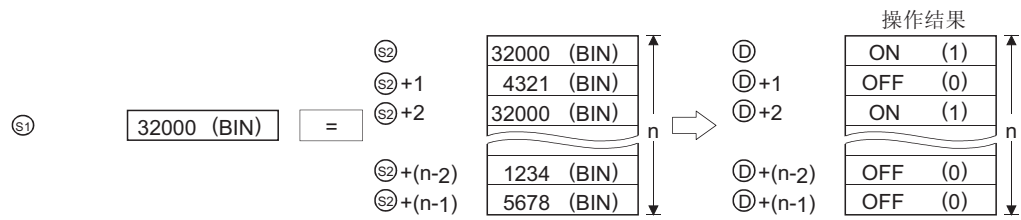
设定数据	含义	数据类型
①	被比较的数据, 或存储被比较数据的起始软元件号	BIN 16 位
②	存储比较数据的起始软元件号	BIN 16 位
③	存储比较操作结果的起始软元件号	位
n	被比较的数据块数	BIN 16 位

[功能]

- (1) 从由①指定的软元件号开始的第 n 点 BIN 16 位数据与由②指定的软元件号开始的第 n 点 BIN16 位数据进行比较, 并且将结果依此向前存储到从由③指定的软元件开始的软元件中去。
- (a) 如果满足比较条件, 则③指定的软元件将变为 ON。
- (b) 如果比较条件不满足, 则③指定的软元件将变为 OFF。



- (2) 比较操作在 16 位单元中进行。
- (3) 由①指定的常数范围在-32768 与 32767 之间 (BIN 16-位数据)。



(4) 单个指令的比较操作结果如下：

指令符号	条件	比较操作结果	指令符号	条件	比较操作结果
BKCOMP =	① = ②	ON (1)	BKCOMP ≠	① ≠ ②	OFF (0)
BKCOMP < >	① ≠ ②		BKCOMP < >	① = ②	
BKCOMP >	① > ②		BKCOMP >	① ≤ ②	
BKCOMP ≤	① ≤ ②		BKCOMP ≤	① > ②	
BKCOMP <	① < ②		BKCOMP <	① ≥ ②	
BKCOMP ≥	① ≥ ②		BKCOMP ≥	① < ②	

(5) 如果所有存储在从④开始的 n 点比较结果都为 ON (1)，SM704 (块比较信号) 变为 ON。

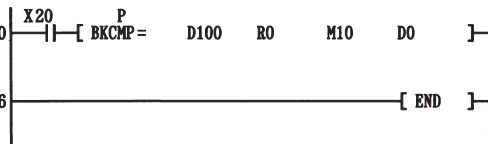
[运行错误]

- (1) 在发生下列运行错误的情况中，错误标志 (SM0) 变为 ON，并且在 SDO 中存储错误代码。
- 从由①、②或④指定的软元件开始 n 点的软元件的范围超过相关软元件。(错误代码：4101)
 - 从由①指定的软元件开始的 n 点软元件的范围与从由④指定的软元件开始的 n 点软元件的范围重叠。(错误代码：4101)
 - 从由②指定的软元件开始的 n 点软元件的范围与从由④指定的软元件开始的 n 点软元件的范围重叠。(错误代码：4101)
 - 从由①指定的软元件开始的 n 点软元件的范围与从由②指定的软元件开始的 n 点软元件的范围重叠。(错误代码：4101)
- (2) 变址修改中的相关错误信息请参考 3.6 节。

[程序示例]

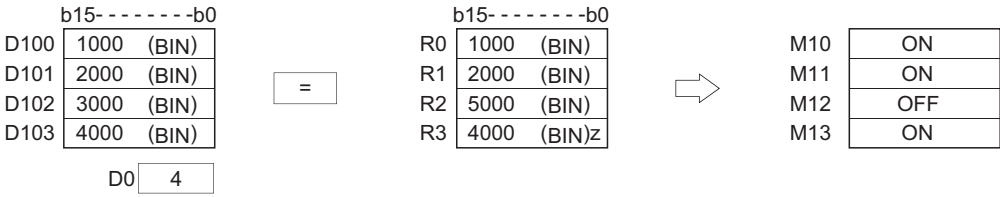
(1) 下列程序在 X20 变 ON 时执行比较操作，将从 D100 地址开始的，与存储在 D0 中的值相等点数的数据，与从 R0 地址开始的，与存储在 D0 中的值相等点数的数据进行比较，并且将结果存储到从 M10 开始的地址中。

[梯形图模式]



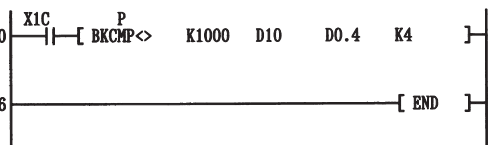
[列表模式]

步	指令	软元件
0	LD	X20
1	BKCMP=P	D100 R0 M10 D0
6	END	



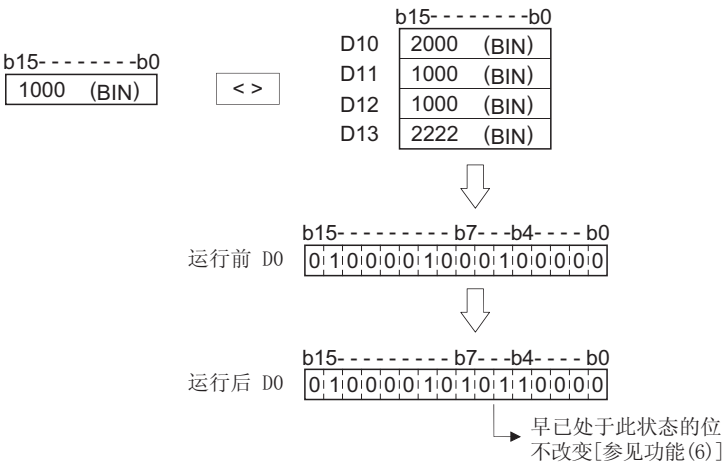
(2) 当 X1C 变为 ON 时，下列程序执行比较操作，将常数 K1000 与从 D10 开始的 4 个点的数据进行比较，并且将结果存储在 D0 的 b4 到 b7 中。

[梯形图模式]



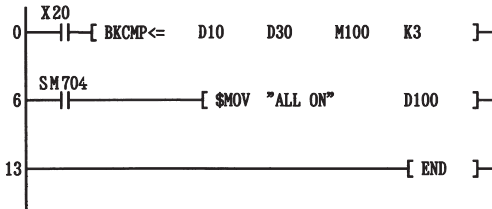
[列表模式]

步	指令	软元件
0	LD	X1C
1	BKCMP<>P	K1000 D10 D0.4 K4
6	END	



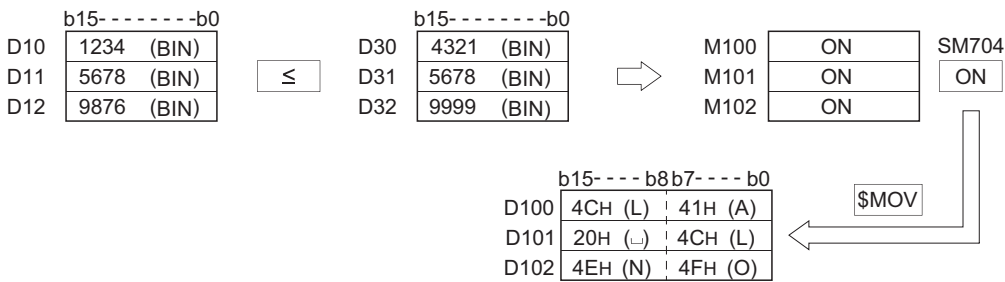
- (3) 当 X20 变为 ON 时, 将从 D10 开始的 3 个点的数据与从 D30 开始的 3 个点的数据进行比较, 并从 M100 向前存储结果。
- 当从 M100 开始的所有软元件都到达 1 “ON” 状态时, 下述程序将字符串 “ALL ON” 传送到 D100 开始的地址中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X20
1	BKCMP<=	D10 D30 M100 K3
6	LD	SM704
7	\$MOV	"ALL ON" D100
13	END	

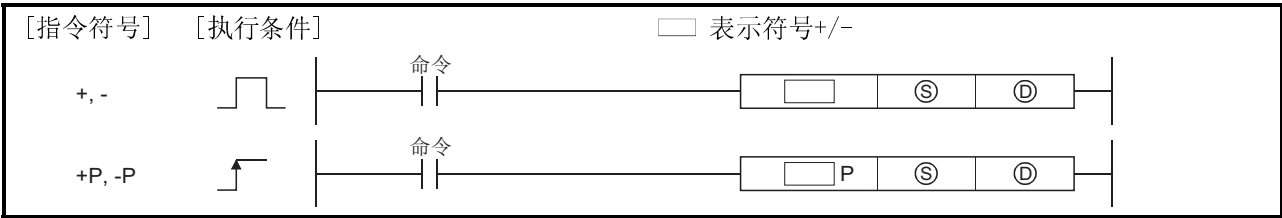


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.2 算术运算指令

6.2.1 BIN 16 位加法和减法运算(+, +P, -与 - P)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	



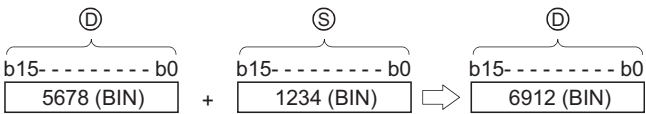
[设定数据]

设定数据	含义	数据类型
Ⓢ	加数、减数或者存储加数、减数的起始软元件号	BIN 16 位
Ⓓ	被加数、被减数或者存储被加数、被减数的起始软元件号	

[功能]



(1) 将由Ⓓ指定的 BIN 16 位数据与由Ⓢ指定的 BIN 16 位数据相加，并将该加法运算的结果存储在由Ⓓ指定的软元件中。



(2) 可以指定给Ⓢ和Ⓓ的值的范围是-32768 到 32767 (BIN，16 位)。

(3) 数据的正负是由它的最高有效位 (b15) 来判定的。

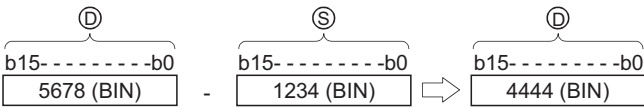
- 0 .. 正
- 1 .. 负

(4) 如果运算结果中产生了下溢或者上溢，则会发生以下情况：
在这种情况下，进位标志不变为 ON。

- K32767 +K2 → K-32767.....如果B15为1，则产生一个负的值.
(H7FFF) (H0002) (H8001)
- K-32768 +K-2 → K32766.....如果B15为0，则产生一个正的值.
(H8000) (HFFFE) (H7FFE)



(1) 从由⑤指定的 BIN 16 位数据中减去由④指定的 BIN 16 位数据，并将该减法运算的结果存储在由⑥指定的软元件中。



(2) 可以指定给⑤和⑥的值的范围是-32768 到 32767 (BIN，16 位)。

(3) 数据的正负是由它的最高有效位 (b15) 来判定的。

- 0 .. 正
- 1 .. 负

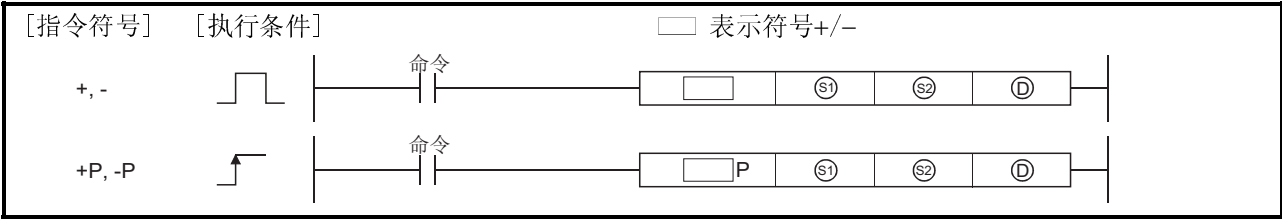
(4) 如果运算结果中产生了下溢或者上溢，则会发生以下情况：
在这种情况下，进位标志不变为 ON。

- K-32768 -K2 → K32766.....如果B15为0，则产生一个正的值
(H8000) (H0002) (H7FFE)
- K32767 -K-2 → K-32767.....如果B15为1，则产生一个负的值
(H7FFF) (H0002) (H8001)

[运行错误]

(1) 不存在与+ (P) 或- (P) 指令相关的运行错误。

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
①	○						○	—	
②	○						○	—	
③	○						—	—	



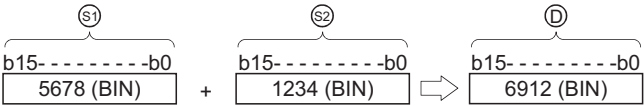
[设定数据]

设定数据	含义	数据类型
①	被加数、被减数或者存储被加数、被减数的起始软元件号	BIN 16 位
②	加数、减数或者存储加数、减数的起始软元件号	
③	存储加法和减法运算结果的起始软元件号	

[功能]



(1) 将由①指定的 BIN 16 位数据与由②指定的 BIN 16 位数据相加，并将结果存储在由③指定的软元件中。



(2) 可以指定给①, ②和③的值的范围是-32768 到 32767 (BIN 16 位)。

(3) 数据的正负是由它的最高有效位 (b15) 来判定的。

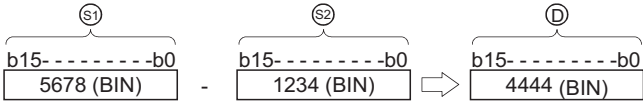
- 0 .. 正
- 1 .. 负

(4) 如果运算结果中产生了下溢或者上溢，则会发生以下情况：
在这种情况下，进位标志不变为 ON。

- K32767 +K2 → K-32767.....如果B15为1，则产生一个负的值
(H7FFF) (H0002) (H8001)
- K-32768 +K-2 → K32766.....如果B15为0，则产生一个正的值
(H8000) (HFFFE) (H7FFE)



(1) 从由(S2)指定的 BIN 16 位数据中减去由(S1)指定的 BIN 16 位数据，并将减法运算的结果存储在由(D)指定的软元件中。



(2) 可以指定给(S1), (S2)和(D)的值的范围是-32768 到 32767 (BIN 16 位)。

(3) 数据的正负是由它的最高有效位 (b15) 来判定的。

- 0 .. 正
- 1 .. 负

(4) 如果运算结果中产生了下溢或者上溢，则会发生以下情况：
在这种情况下，进位标志不变为 ON。

- K-32768 -K2 → K32766.....如果B15为0，则产生一个正的
(H8000) (H0002) (H7FFE)
- K32767 -K-2 → K-32767.....如果B15为1，则产生一个负的
(H7FFF) (H0002) (H8001)

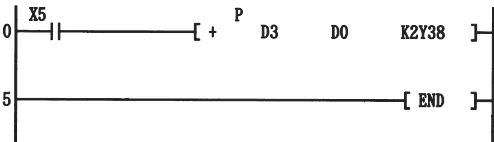
[运行错误]

(1) 不存在与+ (P) 或- (P) 指令相关的运行错误。

[程序示例]

(1) 当 X5 变为 ON 时，以下程序将 D3 的内容与 D0 的内容相加，并将结果输出到 Y38 到 Y3F。

[梯形图模式]

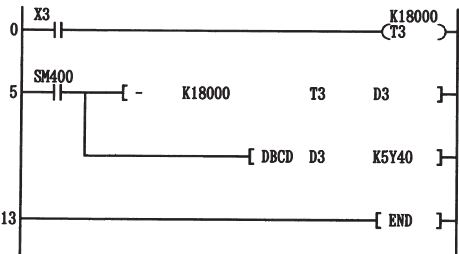


[列表模式]

步	指令	软元件
0	LD	X5
1	+P	D3 D0
		K2Y38
5	END	

(2) 以下程序将定时器 T3 的设定值与当前值之间的差值输出到 Y40 到 Y53。

[梯形图模式]



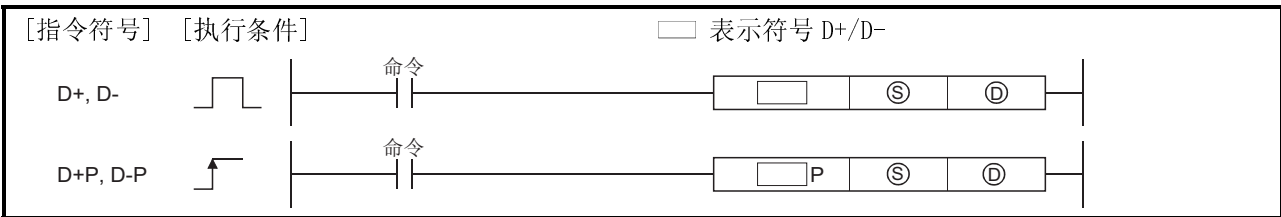
[列表模式]

步	指令	软元件
0	LD	X3
1	OUT	T3
		K18000
5	LD	SM400
6	-	K18000
		D3
10	DBCD	D3
		K5Y40
13	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.2.2 BIN 32 位加法和减法运算(D+, D+P, D-与 D-P)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	



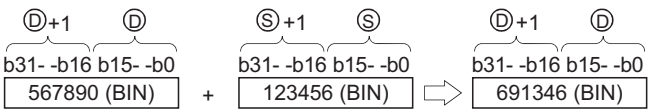
[设定数据]

设定数据	含义	数据类型
Ⓔ	加数、减数或者存储加数、减数的起始软元件号	BIN 32 位
Ⓕ	被加数、被减数或者存储被加数、被减数的起始软元件号	

[功能]

D+

(1) 将由Ⓕ指定的 BIN32 位数据与由Ⓔ指定的 BIN 32 位数据相加，并将该加法运算的结果存储在由Ⓕ指定的软元件中。



(2) 可以指定给Ⓔ和Ⓕ的值的范围是-2147483648 到 2147483647 (BIN 32 位)。

(3) 数据的正负是根据它的最高有效位 (b31) 来判定的。

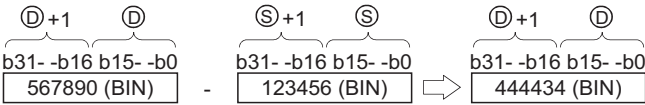
- 0 .. 正
- 1 .. 负

(4) 如果运算结果产生了下溢或者上溢，则会发生以下情况：
在这种情况下，进位标志不变为 ON。

- K2147483647 +K2 → K-2147483647.....因为b31为1，所以值为负。
(H7FFFFFFF) (H2) (H80000001)
- K-2147483648 +K-2 → K2147483646.....因为b31为0，所以值为正。
(H80000000) (HFFFE) (H7FFFFFFE)

D-

(1) 从由⑤指定的 BIN 32 位数据中减去由④指定的 BIN 32 位数据，并将该减法运算的结果存储在由⑥指定的软元件中。



(2) 可以指定给④和⑤的值的范围是-2147483648 到 2147483647 (BIN 32 位)。

(3) 数据的正负是根据它的最高有效位 (b31) 来判定的。

- 0 .. 正
- 1 .. 负

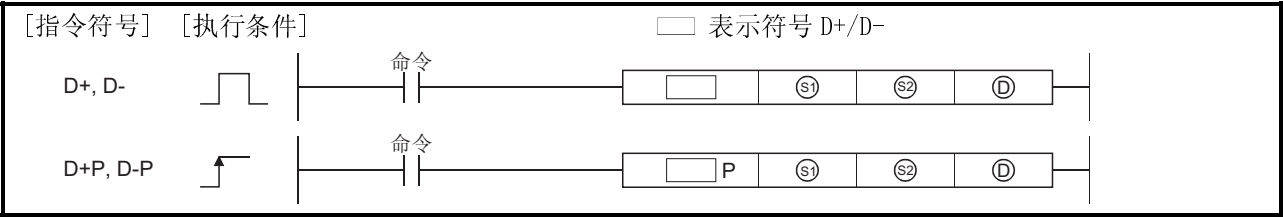
(4) 如果运算结果产生了下溢或者上溢，则会发生以下情况：
在这种情况下，进位标志不变为 ON。

- K-2147483648 -K2 → K2147483646.....因为b31为0，所以值为正。
(H80000000) (H2) (H7FFFFFFE)
- K2147483647 -K-2 → K-2147483647.....因为b31为1，所以值为负。
(H7FFFFFFF) (HFFFE) (H80000001)

[运行错误]

(1) 不存在与+ (P) 或- (P) 指令相关的运行错误。

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
Ⓐ	○						○	—	
Ⓑ	○						○	—	
Ⓒ	○						—	—	



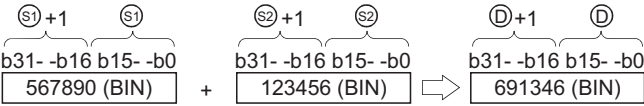
[设定数据]

设定数据	含义	数据类型
Ⓐ	被加数、被减数或者存储被加数、被减数的起始软元件号	BIN 32 位
Ⓑ	加数、减数或者存储加数、减数的起始软元件号	
Ⓒ	存储加法、减法操作结果的起始软元件号	

[功能]

D+

(1) 将由Ⓐ指定的 BIN 32 位数据与由Ⓑ指定的 BIN 32 位数据相加并将其结果存储在由Ⓒ指定的软元件中。



(2) 可以指定给Ⓐ, Ⓑ和Ⓒ的值的范围是-2147483648 到 2147483647 (BIN 32 位)。

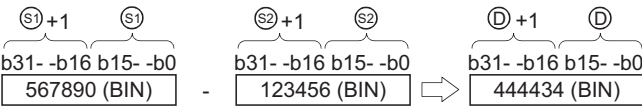
(3) 数据的正负是根据它的最高有效位 (b31) 来判定的。

(4) 如果运算结果中产生了下溢或者上溢，则会发生以下情况:在这种情况下，进位标志不变为 ON。

- K2147483647 +K2 → K-2147483647.....因为b31为0，所以值为正。
(H7FFFFFFF) (H2) (H80000001)
- K-2147483648 +K-2 → K2147483646.....因为b31为1，所以值为负。
(H80000000) (HFFFE) (HFFFE)

D-

(1) 从由⑤₂指定的 BIN 32 位数据中减去由⑤₁指定的 BIN 32 位数据，并将该减法运算的结果存储在由⑤₃指定的软元件中。



(2) 可以指定给⑤₁, ⑤₂和⑤₃的值的范围是-2147483648 到 2147483647 (BIN 32 位)。

(3) 数据的正负是根据它的最高有效位 (b31) 来判定的。

- 0 .. 正
- 1 .. 负

(4) 如果运算结果产生了下溢或者上溢，则会发生以下情况。在这种情况下，进位标志不变为 ON。

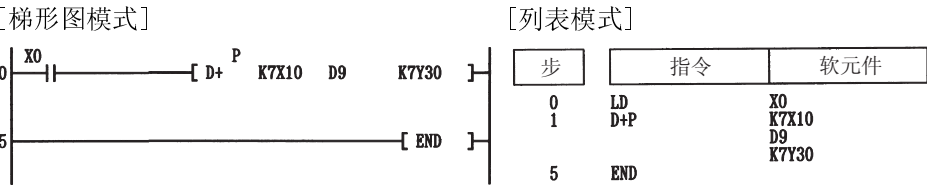
- K-2147483648 -K2 → K2147483646.....因为b31为0，所以值为正。
(H80000000) (H2) (H7FFFFFFE)
- K2147483647 -K-2 → K-2147483647.....因为b31为1，所以值为负。
(H7FFFFFFF) (HFFFE) (HFFFE)

[运行错误]

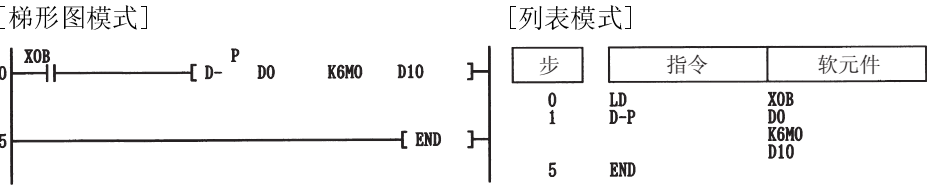
(1) 不存在与+ (P) 或- (P) 指令相关的运行错误。

[程序示例]

(1) 当 X0 变为 ON 时，以下程序将来自 X10 到 X2B 的 28 位数据与 D9 和 D10 上的数据相加，并将运算结果输出到 Y30 到 Y4B。



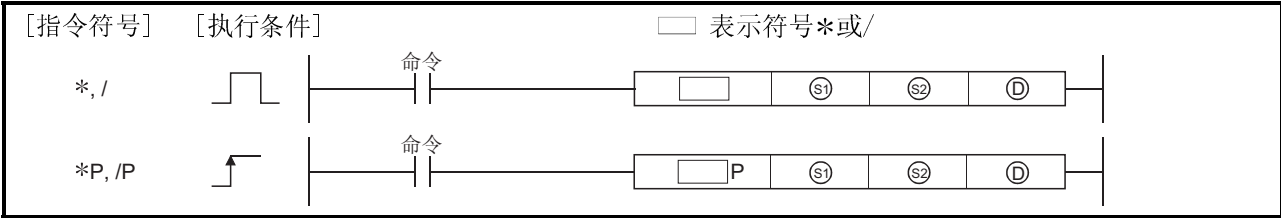
(2) 当 XB 变为 ON 时，以下程序从 D0 和 D1 上的数据中减去来自 M0 到 M23 的数据，并将运算结果存储在 D10 和 D11 上。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.2.3 BIN 16 位乘法和除法运算(*, *P, /与/P)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MMELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][6]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
①	○						○	—	
②	○						○	—	
③	○						—	—	



[设定数据]

设定数据	含义	数据类型
①	被乘数、被除数或者存储被乘数、被除数的起始软元件号	BIN 16 位
②	乘数、除数或者存储乘数、除数的起始软元件号	
③	存储乘法、除法运算的运算结果的起始软元件号	BIN 32 位

[功能]

*

(1) 将由①指定的 BIN 16 位数据与由②指定的 BIN 16 位数据相乘，并将其结果存储在由③指定的软元件中。

①

b15-----b0

5678 (BIN)

×

②

b15-----b0

1234 (BIN)

⇒

③+1

③

b31--b16

b15--b0

7006652 (BIN)

(2) 如果③是位软元件，那么值由低位开始指定。
例如 K1..... 低 4 位 (b0 到 b3)
 K4..... 低 16 位 (b0 到 b15)
 K8..... 低 32 位 (b0 到 b31)

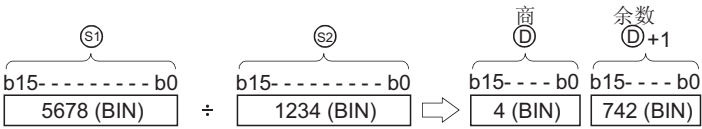
(3) 可以指定给①, ②和③的值的范围是-32768 到 32767 (BIN 16 位)。

(4) ①, ②和③的正负是根据它们的最高有效位 (对①, ②来说，即为 b15；对③，则为 b31) 来判定的。

- 0 .. 正
- 1 .. 负



(1) 将由Ⓢ1指定的 BIN 16 位数据除以由Ⓢ2指定的 BIN 16 位数据，并将其结果存储在ⓐ指定的软元件中。



(2) 如果使用的是字软元件，那么除法运算的结果存储为 32 位，并且将商和余数都存储起来；如果使用的是位软元件，那么结果使用 16 位，并且只存储商。
商..... 存储在低 16 位
余数..... 存储在高 16 位
(只有在使用字软元件时才可以存储)

- (3) 可以指定给Ⓢ1, Ⓢ2和ⓐ的值的范围是-32768 到 32767 (BIN 16 位)。
- (4) Ⓢ1, Ⓢ2和ⓐ的值的正负是根据它们的最高有效位 (对Ⓢ1, Ⓢ2和ⓐ来说，即为 b15) 来判定的。
- 0 .. 正
 - 1 .. 负

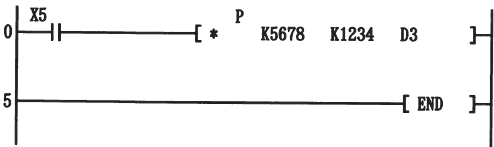
[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 试图用 0 去除Ⓢ2 (错误代码：4100)。

[程序示例]

- (1) 当 X5 变为 ON 时，以下程序用 “1234” “5678”，并将结果存储在 D3 和 D4 中。

[梯形图模式]

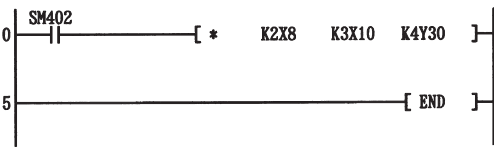


[列表模式]

步	指令	软元件
0	LD	X5
1	*P	K5678 K1234 D3
5	END	

- (2) 以下程序用 X10 到 X1B 上的 BIN 数据去除 X8 到 XF 上的 BIN 数据，并将该除法运算的结果输出到 Y30 到 Y3F。

[梯形图模式]

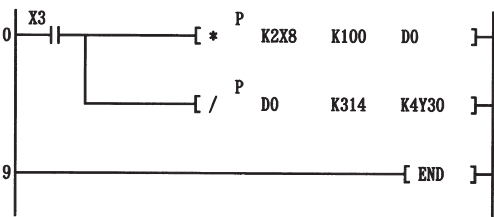


[列表模式]

步	指令	软元件
0	LD	SM402
1	*	K2X8 K3X10 K4Y30
5	END	

- (3) 当 X3 为 ON 时，以下程序将 X8 到 XF 上的数据除以 3.14 的结果值输出到 Y30 到 Y3F。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X3
1	*P	K2X8 K100 D0
5	/P	D0 K314 K4Y30
9	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.2.4 BIN 32 位乘法和除法运算 (D*, D*/P, D/与 D/P)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
①	○			○				—	
②	○			○				—	
④	○			—				—	

[指令符号]	[执行条件]	□ 表示符号 D*或 D/			
D*, D/		命令	□	①	② ④
D*/P, D/P		命令	□P	①	② ④

[设定数据]

设定数据	含义	数据类型
①	被乘数、被除数或者存储被乘数、被除数的起始软元件号	BIN 32 位
②	乘数、除数或者存储乘数、除数的起始软元件号	
④	存储乘法、除法运算的运算结果的起始软元件号	BIN 64 位

[功能]

D*

(1) 将由①指定的 BIN 32 位数据与由②指定的 BIN 32 位数据相乘并将其结果存储在由④指定的软元件中。



(2) 如果④是位软元件，那么将只考虑乘法运算结果的低 32 位，而高 32 位不会被指定。

例如 K1 低 4 位 (b0 到 b3)
 K4 低 16 位 (b0 到 b15)
 K8 低 32 位 (b0 到 b31)

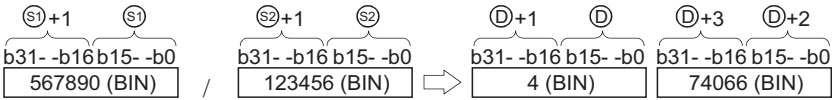
对乘法运算的结果来说，如果该位软元件的高 32 位也需要考虑，那么首先将该数据结果暂存在一个字软元件中，然后将字软元件中的数据传输给指定 (④+2) 和 (④+3) 的数据的位软元件中。

(3) 可以指定给①, ②和④的值的范围是-2147483648 到 2147483647 (BIN 32 位)。

- (4) $\textcircled{S1}$, $\textcircled{S2}$ 和 \textcircled{D} 的值的正负是根据它们的最高有效位(对 $\textcircled{S1}$, $\textcircled{S2}$ 来说, 即为 b31; 对 \textcircled{D} , 则为 b63)来判定的。
- 0 .. 正
 - 1 .. 负

D/

- (1) 用由 $\textcircled{S1}$ 指定的 BIN 32 位数据除以由 $\textcircled{S2}$ 指定的 BIN 32 位数据, 并将其结果存储在 \textcircled{D} 指定的软元件中。



- (2) 如果使用的是字软元件, 那么除法运算的结果存储为 64 位, 并且将商和余数都存储起来; 如果使用的是位软元件, 那么结果使用 32 位, 并且只存储商。
- 商 存储在低 16 位
余数 存储在高 16 位
(只有在使用字软元件时才可以存储)
- (3) 可以指定给 $\textcircled{S1}$, $\textcircled{S2}$ 和 \textcircled{D} 的值的范围是-2147483648 到 2147483647 (BIN 32 位)。
- (4) $\textcircled{S1}$, $\textcircled{S2}$, \textcircled{D} 和 $\textcircled{D}+2$ 的值的正负是根据它们的最高有效位 (b31) 来判定的。
(符号与商和余数一起使用)
- 0 .. 正
 - 1 .. 负

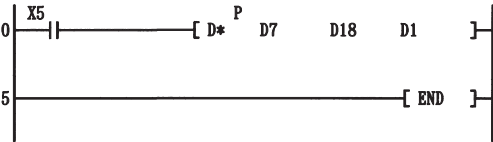
[运行错误]

- (1) 在发生以下运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 试图用 0 去除 $\textcircled{S2}$ (错误代码: 4100)。

[程序示例]

- (1) 当 X5 为 ON 时, 以下程序将 D7 和 D8 中的 BIN 数据除以 D18 和 D19 中的 BIN 数据, 并将结果存储在 D1 到 D4 中。

[梯形图模式]

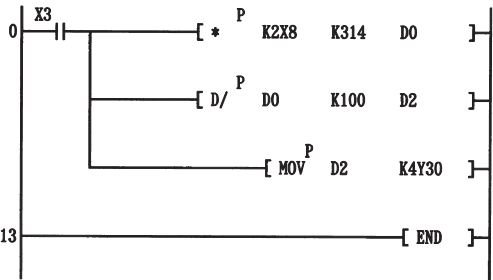


[列表模式]

步	指令	软元件
0	LD	X5
1	D*P	D7 D18 D1
5	END	

- (2) 当 X3 为 ON 时, 以下程序将 X8 到 XF 中的数据乘以 3.14 所得到的结果输出到 Y30 到 Y3F。

[梯形图模式]



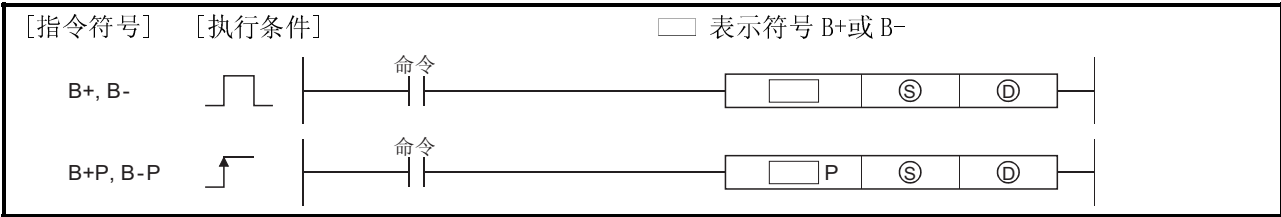
[列表模式]

步	指令	软元件
0	LD	X3
1	*P	K2X8 K314 D0
5	D/P	D0 K100 D2
10	MOV	D2 K4Y30
13	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.2.5 BCD 4 位加法和减法运算 (B+, B+P, B-与 B-P)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[] []		特殊功能模 块 U[] G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	



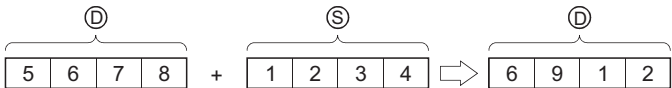
[设定数据]

设定数据	含义	数据类型
⑤	加数、减数或者存储加数、减数的软元件的起始号	BCD 4 位
⑥	被加数、被减数或者存储被加数、被减数的软元件的起始号	

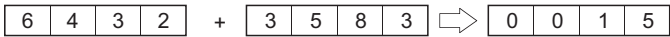
[功能]

B+

- (1) 将⑥指定的 BCD 4 位数据与⑤指定的 BCD 4 位数据相加，并将该加法运算的结果存储在⑥指定的软元件中。



- (2) 可以指定给⑤和⑥的值的范围是 0 到 9999 (BCD 4 位)。
- (3) 如果加法运算的结果超出了 9999，那么其高位将被忽略。在这种情况下，进位标志不变为 ON。



- (2) 可以指定给⑤和⑥的值的范围是 0 到 9999 (BCD 4 位)。
- (3) 如果减法运算产生了下溢，则会导致以下情况：
在这种情况下，进位标志不变为 ON。

0	0	0	1
---	---	---	---

 -

0	0	0	3
---	---	---	---

 ⇨

9	9	9	8
---	---	---	---

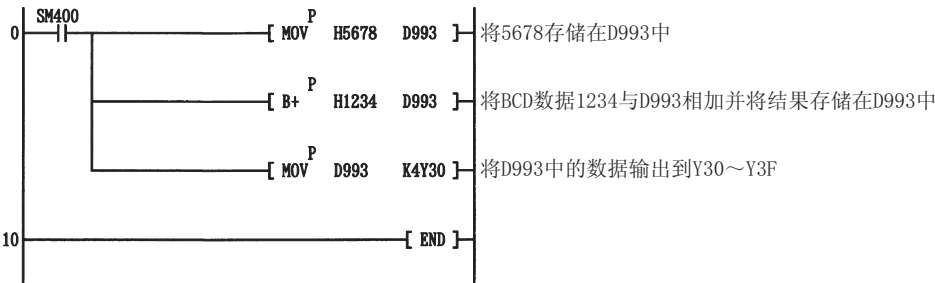
[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- ⑤或⑥上的 BCD 数据超出了 0 到 9999 的范围。(错误代码：4100)

[程序示例]

- (1) 以下程序将 BCD 数据 5678 与 1234 相加，并将结果存储在 D993 中，同时将它输出到 Y30 到 Y3F。

[梯形图模式]

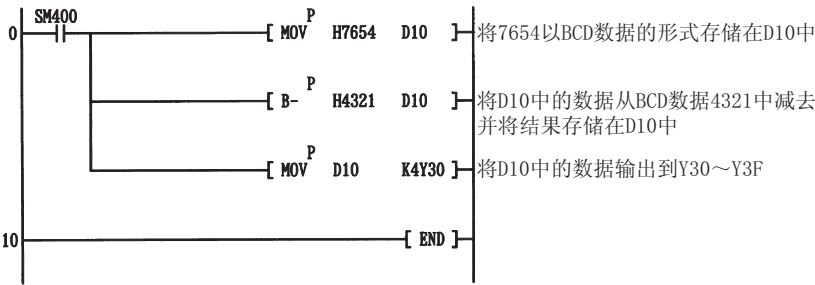


[列表模式]

步	指令	软元件
0	LD	SM400
1	MOVP	H5678 D993
4	B+P	H1234 D993
7	MOVP	D993 K4Y30
10	END	

- (2) 以下程序将 BCD 数据 4321 从 7654 中减去，并将结果存储在 D10 中，同时将它输出到 Y30 到 Y3F。

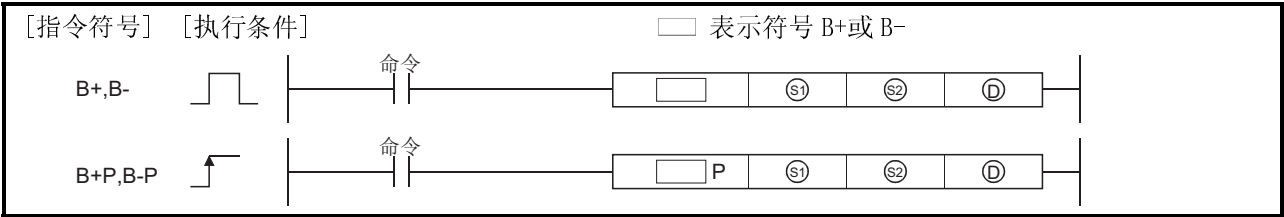
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	MOVP	H7654 D10
4	B-P	H4321 D10
7	MOVP	D10 K4Y30
10	END	

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
Ⓔ1	○						○	—	
Ⓔ2	○						○	—	
Ⓓ	○						—	—	



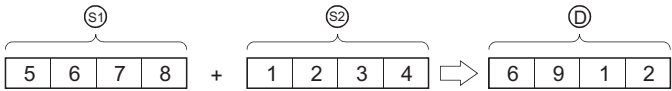
[设定数据]

设定数据	含义	数据类型
Ⓢ1	被加数、被减数或者存储被加数、被减数的起始软元件号	BCD 4 位
Ⓢ2	加数、减数或者存储加数、减数的起始软元件号	
Ⓓ	存储加法、减法运算结果的起始软元件号	

[功能]

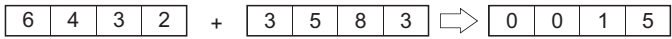
B+

(1) 将Ⓢ1指定的 BCD 4 位数据与Ⓢ2指定的 BCD 4 位数据相加，并将该加法运算的结果存储在Ⓓ指定的软元件中。



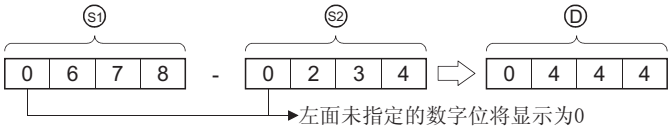
(2) 可以指定给Ⓢ1, Ⓢ2和Ⓓ的值的范围是 0 到 9999 (BCD 4 位)。

(3) 如果加法运算的结果超出了 9999，那么其高位将被忽略。
在这种情况下，进位标志不变为 0N。



B-

(1) 从Ⓢ1指定的 BCD 4 位数据中减去Ⓢ2指定的 BCD 4 位数据，并将该减法运算的结果存储在Ⓓ指定的软元件中。



(2) 可以指定给⑤, ⑥和⑦的值的范围是 0 到 9999 (BCD 4 位)。

(3) 如果减法运算产生了下溢，则会导致以下情况：
在这种情况下，进位标志不变为 ON。

0001

-

0003

⇒

9998

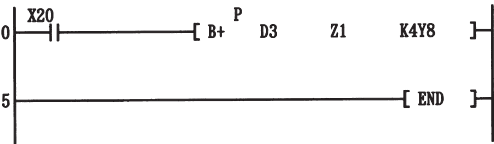
[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- ⑤, ⑥或⑦上的 BCD 数据超出了 0 到 9999 的范围。(错误代码：4100)

[程序示例]

- (1) 当 X20 变为 ON 时，以下程序将 D3 中的 BCD 数据与 Z1 上的 BCD 数据相加，并将其结果输出到 Y8 到 Y17。

[梯形图模式]

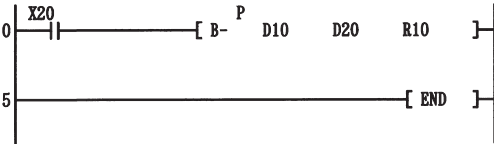


[列表模式]

步	指令	软元件
0	LD	X20
1	B+P	D3 Z1 K4Y8
5	END	

- (2) 当 X20 变为 ON 时，以下程序将 D20 中的 BCD 数据从 D10 上的 BCD 数据中减去，并将其结果存储在 R10 中。

[梯形图模式]



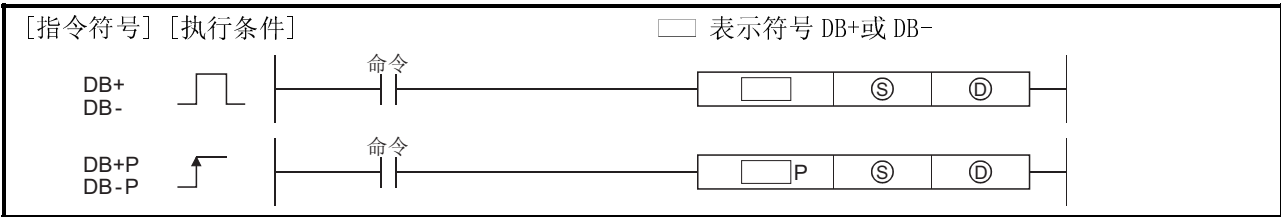
[列表模式]

步	指令	软元件
0	LD	X20
1	B-P	D10 D20 R10
5	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.2.6 BCD 8 位加法和减法运算 (DB+, DB+P, DB-与 DB-P)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[] []		变址寄存器 Zn	其它
	位	字		位	字		
⑤				○			—
⑥				○			—



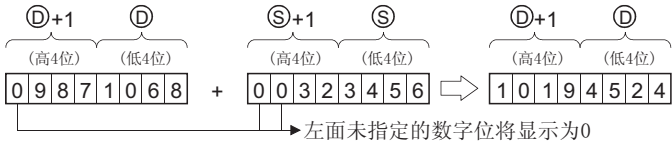
[设定数据]

设定数据	含义	数据类型
⑤	加数、减数或者存储加数、减数的起始软元件号	BCD 8 位
⑥	被加数、被减数或者存储被加数、被减数的起始软元件号	

[功能]

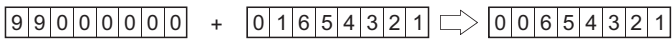
DB+

(1) 将⑥指定的 BCD 8 位数据与⑤指定的 BCD 8 位数据相加，并将该加法运算的结果存储在⑥指定的软元件中。



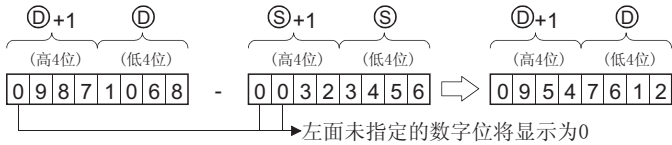
(2) 可以指定给⑤和⑥的值的范围是 0 到 99999999 (BCD 8 位)。

(3) 如果加法运算的结果超出了 99999999，那么其高位将被忽略。
在这种情况下，进位标志不变为 0N。



DB-

(1) 从⑥指定的 BCD 8 位数据中减去⑤指定的 BCD 8 位数据，并将该减法运算的结果存储在⑥指定的软元件中。



- (2) 可以指定给⑤和⑥的值的范围是 0 到 99999999 (BCD 8 位)。
- (3) 如果减法运算产生了下溢，则会导致以下情况：
在这种情况下，进位标志不变为 ON。

12345678

-

12345679

⇒

99999999

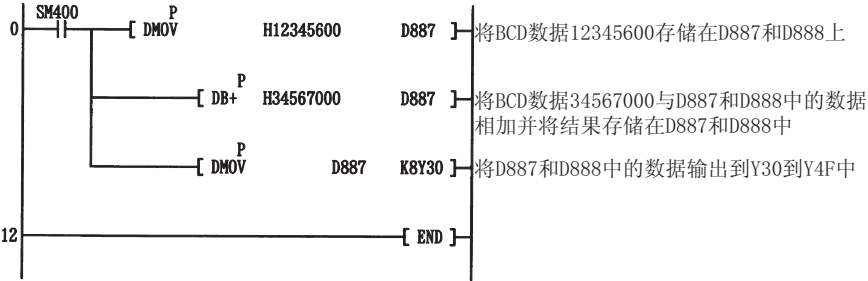
[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
 - ⑤或⑥上的 BCD 数据超出了 0 到 99999999 的范围。(错误代码：4100)

[程序示例]

- (1) 以下程序将 BCD 数据 12345600 与 34567000 相加，并将结果存储在 D887 和 D888 中，同时将它们输出到 Y30 到 Y4F。

[梯形图模式]

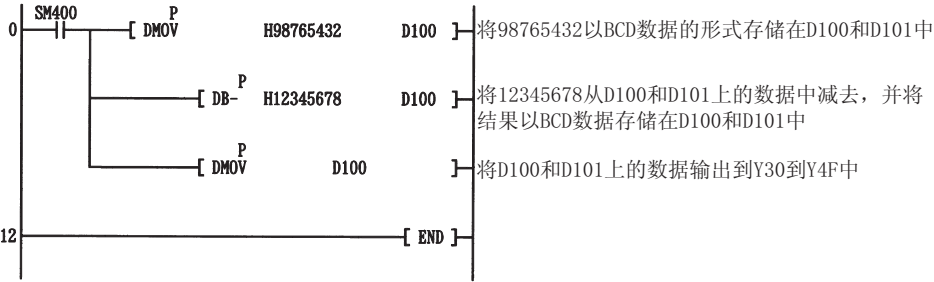


[列表模式]

步	指令	软元件
0	LD	SM400
1	DMOVP	H12345600 D887
5	DB+P	H34567000 D887
9	DMOVP	D887 K8Y30
12	END	

- (2) 以下程序从 12345678 中减去 BCD 数据 98765432，并将结果存储在 D100 和 D101 中，同时将它们输出到 Y30 到 Y4F。

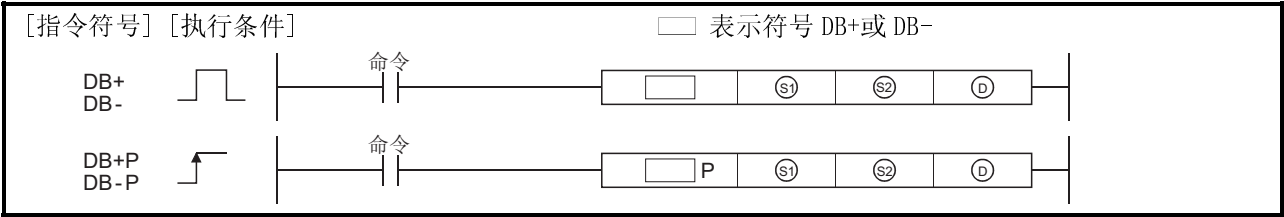
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	DMOVP	H98765432 D100
5	DB-P	H12345678 D100
9	DMOVP	D100 K8Y30
12	END	

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
①	○						○	—	
②	○						○	—	
③	○						—	—	



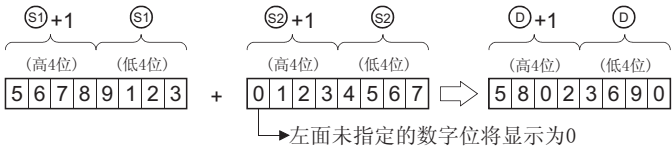
[设定数据]

设定数据	含义	数据类型
Ⓔ1	被加数、被减数或者存储被加数、被减数的起始软元件号	BCD 8 位
Ⓔ2	加数、减数或者存储加数、减数的起始软元件号	
Ⓓ	存储加数、减数的起始软元件号	

[功能]

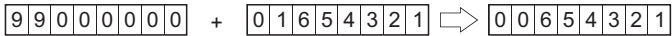
DB+

(1) 将Ⓔ1指定的 BCD 8 位数据与Ⓔ2指定的 BCD 8 位数据相加，并将该加法运算的结果存储在Ⓓ指定的软元件中。



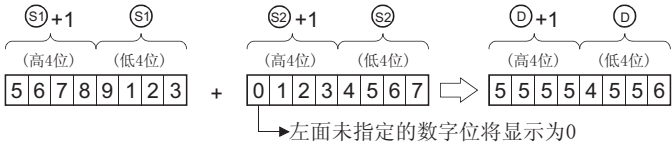
(2) 可以指定给Ⓔ1, Ⓔ2和Ⓓ的值的范围是 0 到 99999999 (BCD 8 位)。

(3) 如果加法运算的结果超出了 99999999，那么其高位将被忽略。
在这种情况下，进位标志不变为 ON。



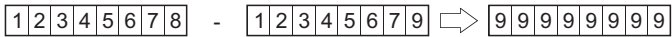
DB-

(1) 从Ⓔ指定的 BCD 8 位数据中减去Ⓕ指定的 BCD 8 位数据，并将该减法运算的结果存储在Ⓖ指定的软元件中。



(2) 可以指定给Ⓔ, Ⓕ和Ⓖ的值的范围是 0 到 99999999 (BCD 8 位)。

(3) 如果减法运算产生了下溢，则会导致以下情况：
在这种情况下，进位标志不变为 0N。



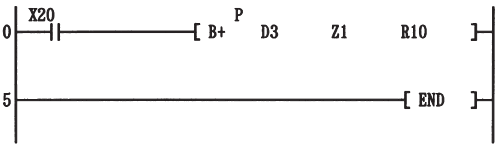
[运行错误]

(1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 0N，并且错误代码存储在 SD0 中。
● Ⓔ, Ⓕ或Ⓖ上的 BCD 数据超出了 0 到 99999999 的范围。(错误代码: 4100)

[程序示例]

(1) 当 X20 变为 0N 时，以下程序将 D3、D4 中的 BCD 数据与 Z1、Z2 中的 BCD 数据相加，并将其结果存储在 R10 和 R11 中。

[梯形图模式]



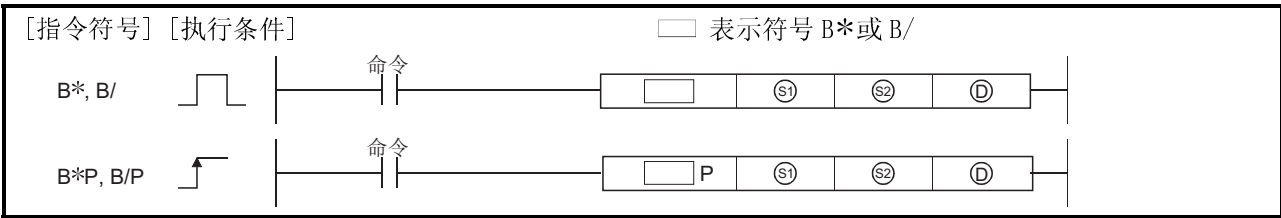
[列表模式]

步	指令	软元件
0	LD	X20
1	B+P	D3 Z1 R10
5	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6. 2. 7 BCD 4 位乘法 and 除法运算 (B*, B*P, B/ 与 B/P)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][6]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
①	○						○	—	
②	○						○	—	
③	○						—	—	



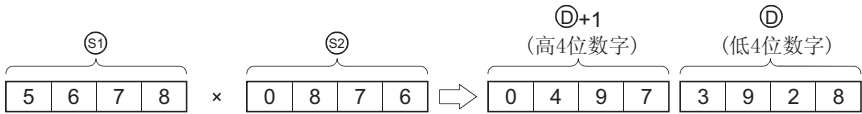
[设定数据]

设定数据	含义	数据类型
①	被乘数、被除数或者存储被乘数、被除数的起始软元件号	BCD 4 位
②	乘数、除数或者存储乘数、除数的起始软元件号	
③	存储乘法、除法运算的运算结果的起始软元件号	BCD 8 位

[功能]

B*

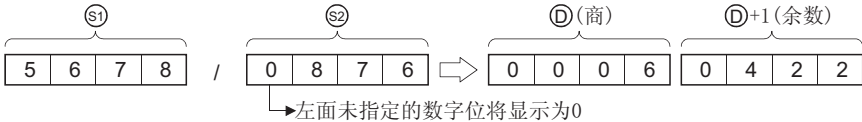
(1) 将①指定的 BCD 数据与②指定的 BCD 数据相乘，并将其结果存储在③指定的软元件中。



(2) 可以指定给①和②的值的范围是 0 到 9999 (BCD 4 位)。

B/

(1) 用①指定的 BCD 数据除以②指定的 BCD 数据，并将其结果存储在③指定的软元件中。



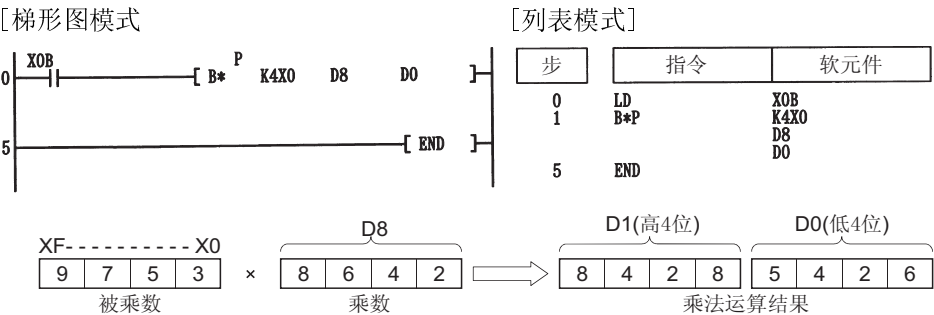
- (2) 使用 32 位数据来存储除法运算的结果：商和余数
- 商 (BCD 4 位) 存储在低 16 位
- 余数 (BCD 4 位) 存储在高 16 位
- (3) 如果已把①指定为位软元件，那么将不存储除法运算产生的余数。

[运行错误]

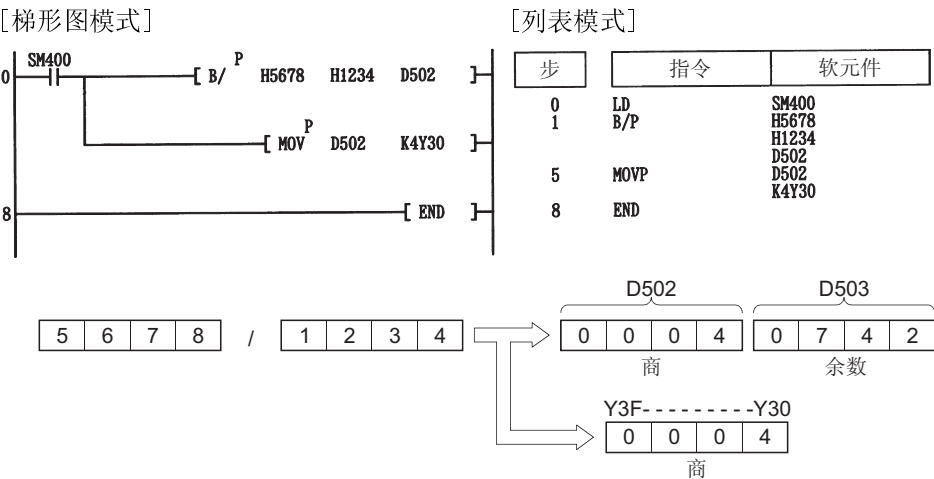
- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- ③或④上的数据超出了 0 到 9999 的范围。(错误代码：4100)
 - 试图用 0 去除②。(错误代码：4100)

[程序示例]

- (1) 当 X0B 变为 ON 时，以下程序将 X0 到 XF 中的 BCD 数据与 D8 中的 BCD 数据相乘，并将运算结果存储在 D0 和 D1 中。



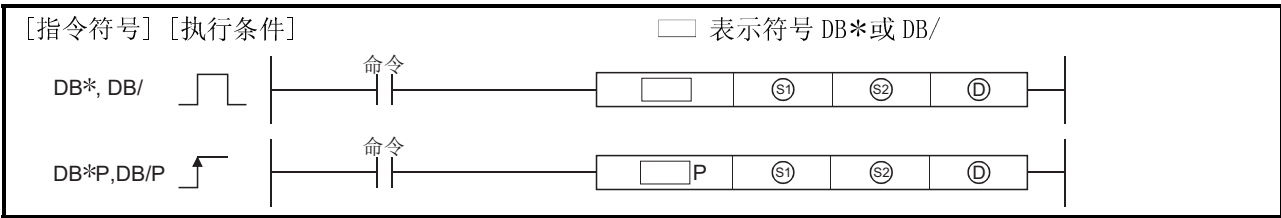
- (2) 以下程序用 5678 除以 BCD 数据 1234，并将其结果存储在 D502 和 D503 中，同时将商输出到 Y30 到 Y3F。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.2.8 BCD 8 位乘法 and 除法运算 (DB*, DB*/P, DB/与 DB/P)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
Ⓢ1	○			○			—
Ⓢ2	○			○			—
Ⓓ	○			—			—



[设定数据]

设定数据	含义	数据类型
Ⓢ1	被乘数、被除数或者存储被乘数、被除数的起始软元件号	BCD 8 位
Ⓢ2	乘数、除数或者存储乘数、除数的起始软元件号	
Ⓓ	存储乘法、除法运算的运算结果的起始软元件号	BCD 16 位

[功能]

DB*

(1) 将Ⓢ1指定的 BCD 8 位数据与Ⓢ2指定的 BCD 8 位数据相乘，并将其乘积存储在Ⓓ指定的软元件中。

Ⓢ1+1

Ⓢ1

Ⓢ2+1

Ⓢ2

9 9 9 9

9 9 9 9

×

9 9 9 9

9 9 9 9

Ⓓ+3

Ⓓ+2

Ⓓ+1

Ⓓ

9 9 9 9

9 9 9 8

0 0 0 0

0 0 0 1

⇒

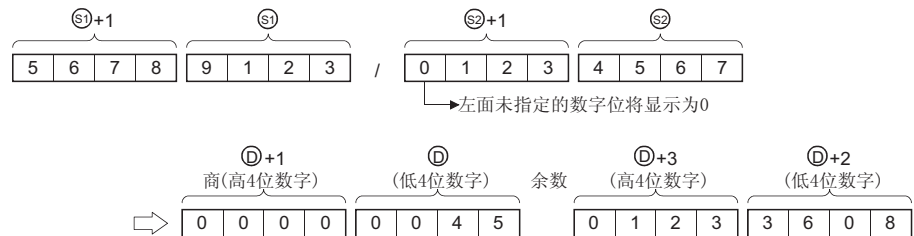
(2) 如果已把Ⓓ指定为位软元件，那么乘积将使用低 8 位数字(低 32 位)，而高 8 位数字(高 32 位)不会被指定。

K1··· 低 1 位数字 (b0 到 b3)
K4··· 低 4 位数字 (b0 到 b15)
K8··· 低 8 位数字 (b0 到 b31)

(3) 可以指定给Ⓢ1和Ⓢ2的值的范围是 0 到 99999999 (8 位 BCD)。

DB/

(1) 用Ⓔ指定的 8 位 BCD 数据除以Ⓕ指定的 8 位 BCD 数据，并将其结果存储在Ⓖ指定的软元件中。



- (2) 除法运算结果使用 64 位，并以商和余数的形式存储。
商 (BCD 8 位) 存储在低 32 位
余数 (BCD 8 位) 存储在高 32 位
- (3) 如果已把Ⓖ指定为位软元件，那么将不存储除法运算产生的余数。

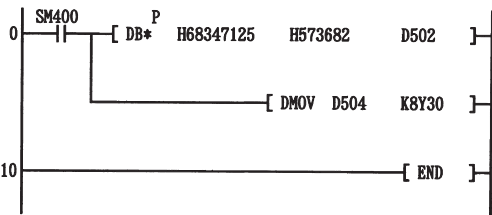
[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- Ⓔ或Ⓕ上的数据超出了 0 到 9999999 的范围。(错误代码: 4100)
 - 试图用 0 去除Ⓕ。(错误代码: 4100)

[程序示例]

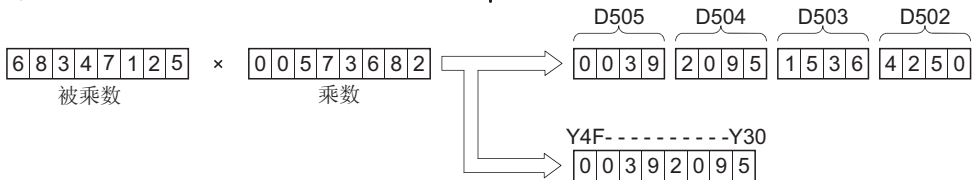
- (1) 以下程序将 BCD 数据 67347125 与 573682 相乘，并将其结果存储在 D502 到 D505 中，同时把它的高 8 位数字输出到 Y30 到 Y4F。

[梯形图模式]



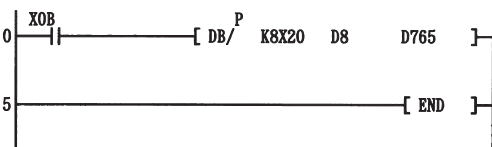
[列表模式]

步	指令	软元件
0	LD	SM400
1	DB*P	H68347125 H573682 D502
7	DMOV	D504 K8Y30
10	END	



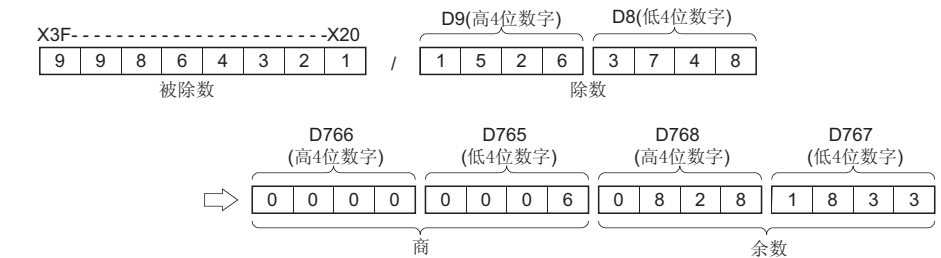
- (2) 当 X0B 变为 ON 时，以下程序用 X20 到 X3F 上的 BCD 数据除以 D8 到 D9 中的 BCD 数据，并将其结果存储在 D765 到 D768 中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0B
1	DB/P	K8X20 D8 D765
5	END	

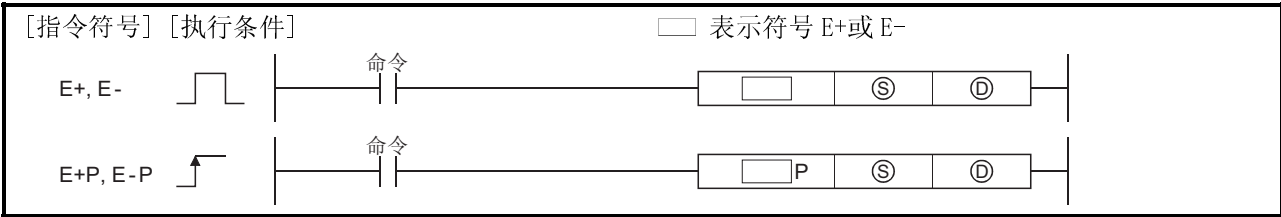


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

※: 序列号的前五位数字是04122或者更大的

6.2.9 浮点数据的加法和减法运算 (E+, E+P, E-与 E-P)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[] []		特殊功能模 块 U[] G[]	变址寄存器 Zn	常量 E	其它
	位	字		位	字				
⑤	—	○		—	○		—	○	—
⑥	—	○		—	○		—	—	—



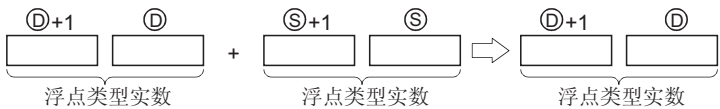
[设定数据]

设定数据	含义	数据类型
Ⓢ	加数、减数或者存储加数、减数的起始软元件号	实数
Ⓓ	被加数、被减数或者存储被加数、被减数的起始软元件号	

[功能]

E+

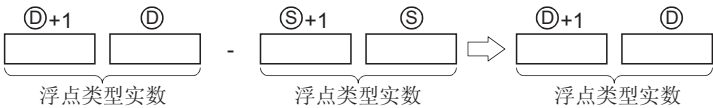
- (1) 将Ⓓ指定的浮点类型实数与Ⓢ指定的浮点类型实数相加，并将它们的和存储在Ⓓ指定的软元件中。



- (2) 可以指定给Ⓢ和Ⓓ的值以及可以存储的值的范围如下：
- $0, \pm 2^{-126} \leq | \text{指定的值 (存储的值)} | < \pm 2^{128}$

E-

(1) 从Ⓓ指定的浮点类型实数中将Ⓢ指定的浮点类型实数减去，并将其结果存储在Ⓓ指定的软元件中。



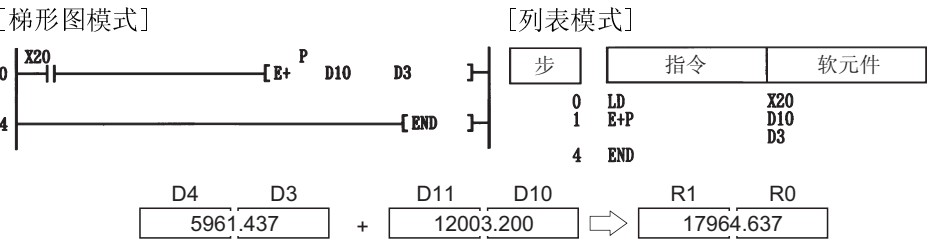
(2) 可以指定给Ⓢ和Ⓓ的值以及可以存储的值的范围如下：
 $0, \pm 2^{-126} \leq \text{指定的值 (存储的值)} < \pm 2^{128}$

[运行错误]

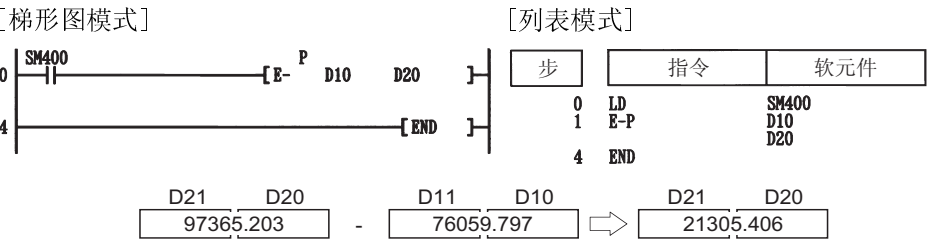
- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 指定软元件的内容或加法, 减法运算的结果不为“0”，或者不在下列范围之内：（错误代码：4100）
 $\pm 2^{-126} \leq \text{指定软元件的内容/运算结果} < \pm 2^{128}$
 - 当指定软元件包含-0 的时候。（错误代码：4100）*

[程序示例]

(1) 当 X20 变为 ON 时，以下程序将 D3 和 D4 中的浮点类型实数与 D10 和 D11 中的浮点类型实数相加，并将其结果存储在 D3 到 D4 中。



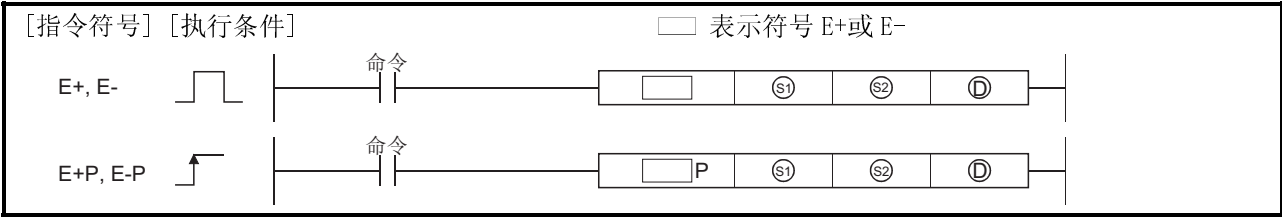
(2) 以下程序从 D20 和 D21 上的浮点类型实数中将 D10 和 D11 中的浮点类型实数减去，并将该减法运算的结果存储在 D20 和 D21 中。



注释

*:如果指定了-0，那么存在不会导致运行错误的 CPU 模块。
有关详细信息，请参阅 3.2.4 节。

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\K[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 E	其它
	位	字		位	字				
Ⓔ1	—	○		—	○		—	○	—
Ⓔ2	—	○		—	○		—	○	—
Ⓓ	—	○		—	○		—	—	—



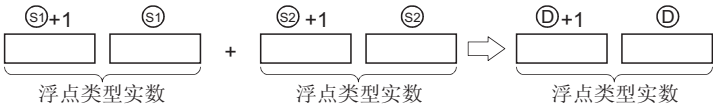
[设定数据]

设定数据	含义	数据类型
Ⓔ1	被加数、被减数或者存储被加数、被减数的起始软元件号	实数
Ⓔ2	加数、减数或者存储加数、减数的起始软元件号	
Ⓓ	存储加法、减法运算结果的起始软元件号	

[功能]

E+

(1) 将Ⓔ1指定的浮点类型实数与Ⓔ2指定的浮点类型实数相加，并将其结果存储在Ⓓ指定的软元件中。

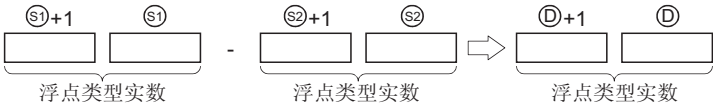


(2) 可以指定给Ⓔ1, Ⓔ2或Ⓓ的值以及可以存储的值的范围如下：

$0, \pm 2^{-126} \leq \text{指定的值 (存储的值)} < \pm 2^{128}$

E-

(1) 从(S2)指定的浮点类型实数中减去(S1)指定的浮点类型实数，并将其结果存储在(D)指定的软元件中。



(2) 可以指定给(S1), (S2)或(D)的值以及可以存储的值的范围如下：

$0, \pm 2^{-126} \leq | \text{指定的值 (存储的值)} | < \pm 2^{128}$

[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 指定软元件的值或加法、减法运算的结果不在下列范围之内：（错误代码：4100）
 $0, \pm 2^{-126} \leq | \text{指定软元件的内容/运算结果} | < \pm 2^{128}$
 - 当指定软元件包含-0 的时候。（错误代码：4100）*

[程序示例]

(1) 当 X20 变为 ON 时，以下程序将 D3 和 D4 中的浮点类型实数与 D10 和 D11 中的浮点类型实数相加，并将其结果输出到 R0 和 R1。

[梯形图模式]

[列表模式]

步	指令	软元件
0	LD	X20
1	E+P	D3 D10 R0
5	END	

D4 D3

5961.437

 +

D11 D10

12003.200

 =>

D4 D3

17964.637

(2) 以下程序将 D20 和 D21 中的浮点类型实数从 D10 和 D11 中的浮点类型实数中减去，并将其结果存储在 D30 和 D31 中。

[梯形图模式]

[列表模式]

步	指令	软元件
0	LD	SM400
1	E-P	D10 D20 D30
5	END	

D11 D10

97365.203

 -

D21 D20

76059.797

 =>

D21 D20

21305.406

注释

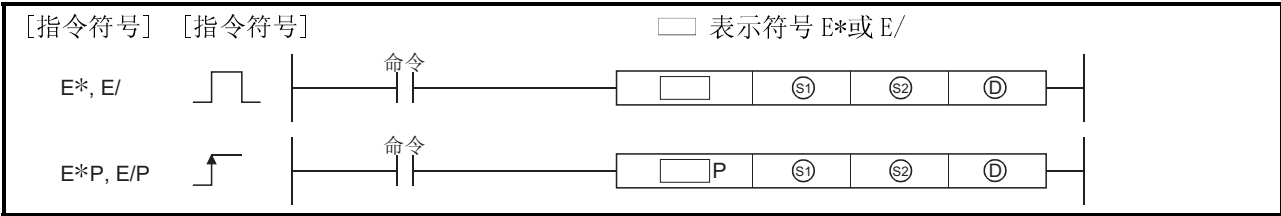
*:如果指定了-0，那么存在不会导致运行错误的 CPU 模块。
有关详细信息，请参阅 3.2.4 节。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

*: 序列号的前五位数字是04122或者更大的

6.2.10 浮点数据的乘法和除法运算(E*, E*/与 E/P)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]G[]	变址寄存器 Zn	常量 E	其它
	位	字		位	字				U
①	—	○		—	○		—	○	—
②	—	○		—	○		—	○	—
③	—	○		—	○		—	—	—



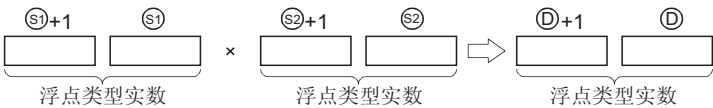
[设定数据]

设定数据	含义	数据类型
①	被乘数、被除数或者存储被乘数、被除数的起始软元件号	实数
②	乘数、除数或者存储乘数、除数的起始软元件号	
③	存储乘法、除法运算的运算结果的起始软元件号	

[功能]

E*

(1) 将①指定的浮点类型实数与②指定的浮点类型实数相乘，并将其乘积存储在③指定的软元件中。

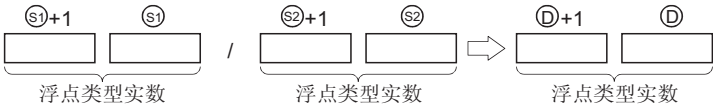


(2) 可以指定给①, ②或③的值以及可以存储的值的范围如下:

$0, \pm 2^{-126} \leq \text{指定的值 (存储的值)} < \pm 2^{128}$

E/

(1) 用⑤1指定的浮点类型实数除以⑤2指定的浮点类型实数，并将其结果存储在⑤D指定的软元件中。



(2) 可以指定给⑤1, ⑤2或⑤D的值以及可以存储的值的范围如下：

$0, \pm 2^{-126} \leq | \text{指定的值 (存储的值)} | < \pm 2^{128}$

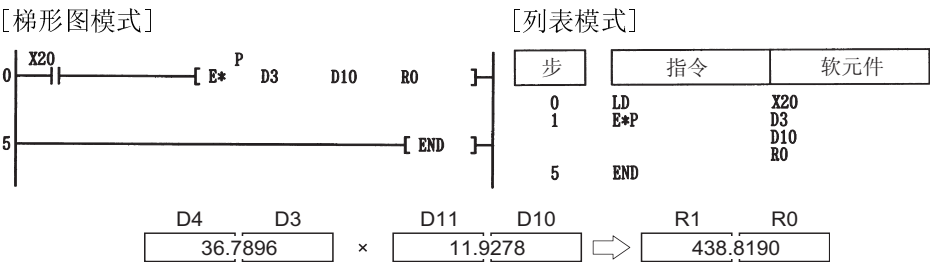
[运行错误]

(1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。

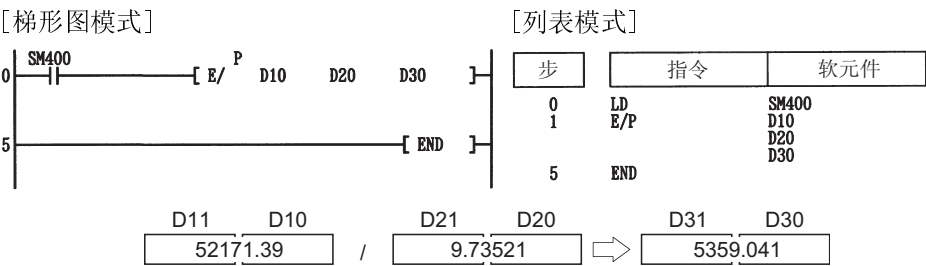
- 指定软元件的值或乘法、除法运算的结果不在下列范围之内：（错误代码：4100）
 $0, \pm 2^{-126} \leq | \text{指定软元件的内容/运算结果} | < \pm 2^{128}$
- 当指定软元件包含-0 的时候。（错误代码：4100）*

[程序示例]

(1) 以下程序将 D3 和 D4 中的浮点类型实数与 D10 和 D11 中的浮点类型实数相乘，并将其结果存储在 R0 和 R1 中。



(2) 以下程序用 D20 和 D21 中的浮点类型实数去除 D10 和 D11 中的浮点类型实数，并将其结果存储在 D30 和 D31 中。



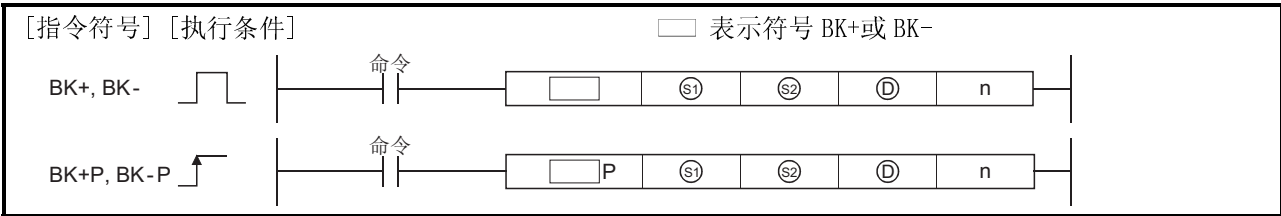
说明

*:如果指定了-0 那么就不会导致运行错误的 CPU 模块是存在的。
有关详细信息，请参阅 3.2.4 节。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.2.11 块加法和减法运算 (BK+, BK+P, BK-与 BK-P)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
①	—	○		—				—	—
②	—	○		—				○	—
③	—	○		—				—	—
n	○	○		○				○	—

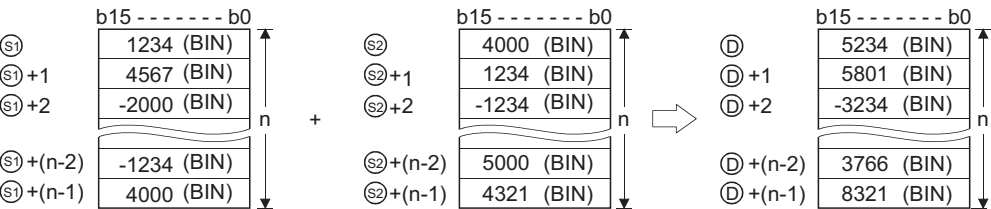


[设定数据]

设定数据	含义	数据类型
①	被加数、被减数或者存储被加数、被减数的起始软元件号	BIN 16 位
②	加数、减数或者存储加数、减数的起始软元件号	
③	存储运算结果的起始软元件号	
n	加/减数据块的数量	

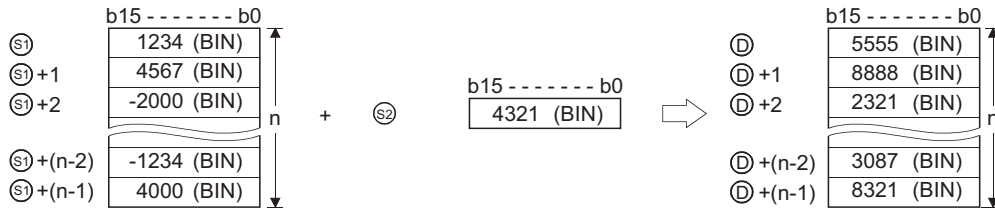
[功能]

- BK+
- (1) 将从①指定的软元件开始的 n 点 BIN 数据与从②指定的软元件开始的 n 点 BIN 数据相加，并将其结果存储在以③指定的软元件中。



- (2) 块加法运算在 16 位单元中进行。

(3) 可以指定给Ⓔ的常数的范围是-32768 到 32767 (BIN16 位)。

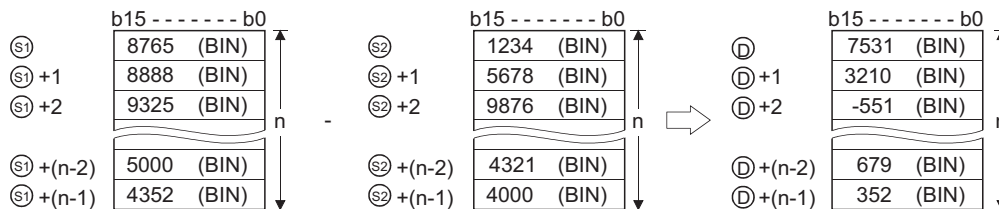


(4) 如果运算结果中产生了下溢或者上溢，则会发生以下情况。
在这种情况下，进位标志不变为 0N。

- K32767 +K2 → K-32767
(H7FFF) (H0002) (H8001)
- K-32767 +K-2 → K32766
(H8000) (HFFFE) (H7FFE)

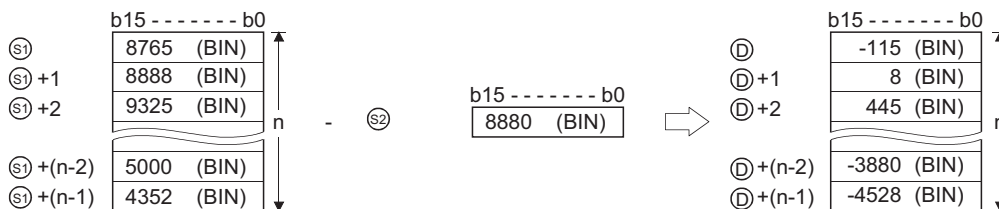
BK-

(1) 从Ⓔ指定的软元件开始的 n 点 BIN 数据中减去从Ⓕ指定的软元件开始的 n 点 BIN 数据，并将其结果存储在以Ⓖ指定的软元件中。



(2) 块减法运算在 16 位单元中进行。

(3) 可以指定给Ⓔ的常数的范围是-32768 到 32767 (BIN16 位)。



(4) 如果运算结果中产生了下溢或者上溢，则会发生以下情况。
在这种情况下，进位标志不变为 0N。

- K-32768 -K2 → K32766
(H8000) (H0002) (H7FFF)
- K32767 -K-2 → -32766
(H8000) (H0002) (H8001)

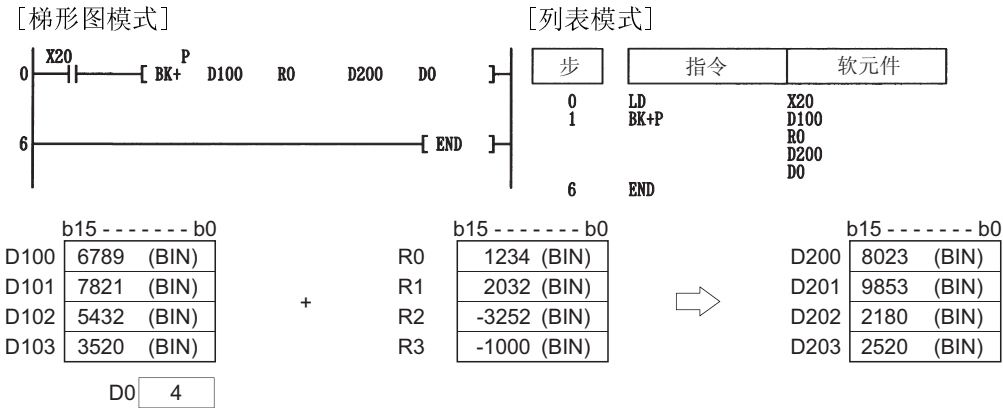
[运行错误]

(1) 在发生以下运行错误的情况中，错误标志(SM0)变为 0N，并且错误代码存储在 SD0 中。

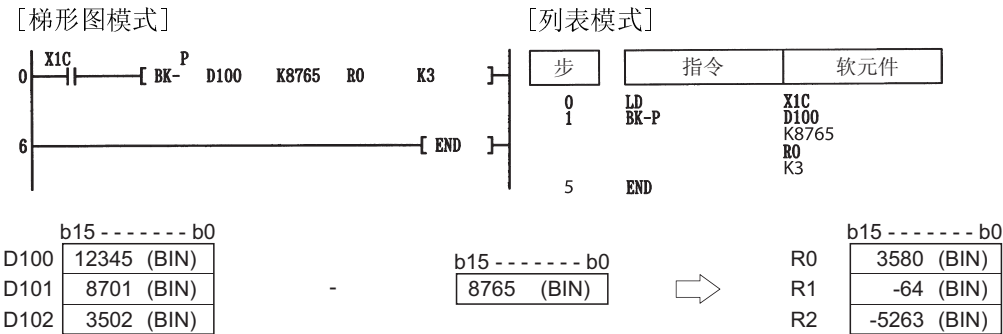
- 从Ⓔ, Ⓕ或Ⓖ软元件起始的 n 位范围超出了该软元件的范围。
- 从Ⓔ指定的软元件开始的 n 点软元件范围与从Ⓖ指定的软元件开始的 n 点软元件范围重叠。
(错误代码: 4101)
- 从Ⓕ指定的软元件开始的 n 点软元件范围与从Ⓖ指定的软元件开始的 n 点软元件范围重叠。
(错误代码: 4101)
- 从Ⓔ指定的软元件开始的 n 点软元件范围与从Ⓕ指定的软元件开始的 n 点软元件范围重叠。
(错误代码: 4101)

[程序示例]

- (1) 当 X20 变为 ON 时，该程序执行下列数据的加法运算：
- 起始于 D100 的软元件号 (存储在 D0 中的值) 中的数据
 - 起始于 R100 的软元件号 (存储在 D0 中的值) 中的数据
- 然后，它将加法运算的结果存储在起始于 D200 的软元件号 (存储在 D0 中的值) 中。



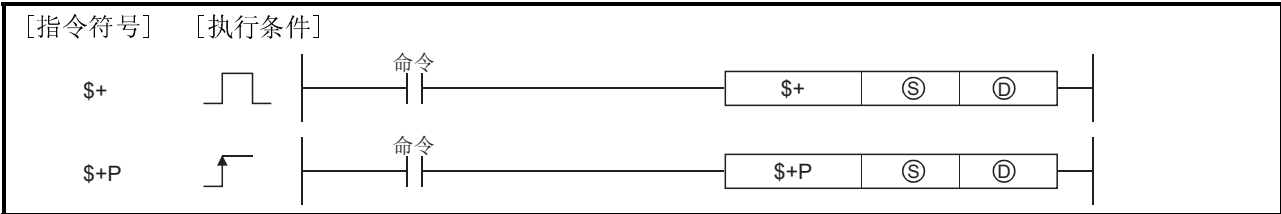
- (2) 当 X1C 变为 ON 时，以下程序从起始于 D100 的三个软元件中存储的数据中减去常数 8765。然后，该程序将减法运算的结果存储在起始于 R0 的三个软元件中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

6.2.12 链接字符串(\$+与\$+P)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 \$	其它
	位	字		位	字				
⑤	—	○		—				○	—
⑥	—	○		—				—	—

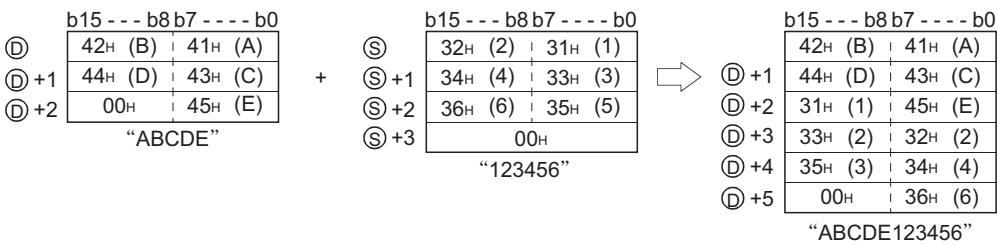


[设定数据]

设定数据	含义	数据类型
Ⓢ	存储数据或被链接数据的起始软元件号	字符串
Ⓓ	存储已完成链接的数据的起始软元件号	

[功能]

- (1) 将从Ⓢ指定的软元件号开始的软元件中存储的字符串数据追加到从Ⓓ指定的软元件中存储的字符串数据的后面，并存储在从Ⓓ指定的软元件号开始的软元件中。
- 该字符串数据对象是指从Ⓓ和Ⓢ指定的软元件号开始到存储“00H”的软元件号为止的软元件中存储的字符串数据。



- (2) 当链接字符串的时候，表示Ⓓ指定的字符串数据的结束的“00H”将被忽略，并且，将Ⓢ指定的字符串数据追加到Ⓓ字符串的最后一个字符之后。

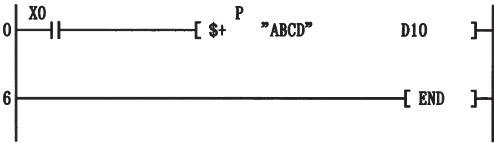
[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志(SM0)变为 0N，并且错误代码存储在 SD 中。
- 不能存储从⑤指定的软元件号到相关软元件的最后一个软元件号之间，已完成链接的整个字符串。(错误代码：4101)
 - ⑤与④指定的字符串的存储软元件号相互重叠。(错误代码：4101)
- (2) 有关变址修改过程中的错误的信息，请参阅 3.6 节。

[程序示例]

(1) 当 X0 变为 0N 时，以下程序将存储在 D10 到 D12 中的字符串与字符串“ABCD”链接起来。

[梯形图模式]

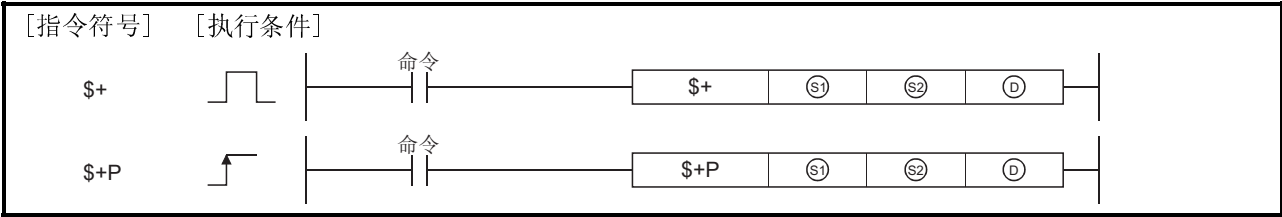


[列表模式]

步	指令	软元件
0	LD	X0
1	\$+P	"ABCD" D10
6	END	



设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G]	变址寄存器 Zn	常量 \$	其它
	位	字		位	字				
①	—	○		—				○	—
②	—	○		—				○	—
③	—	○		—				—	—

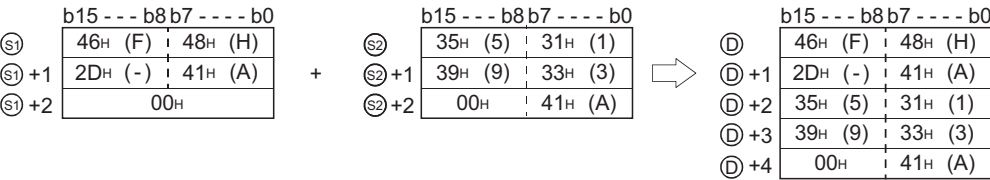


[设定数据]

设定数据	含义	数据类型
Ⓔ1	存储被链接数据的起始软元件号	字符串
Ⓔ2	存储已完成链接的数据的起始软元件号	
Ⓓ	存储链接结果的起始软元件号	

[功能]

- (1) 将从Ⓔ2指定的软元件中存储的字符串数据追加到从Ⓔ1指定的软元件中存储的字符串数据的后面，并存储在从Ⓓ指定的软元件中。



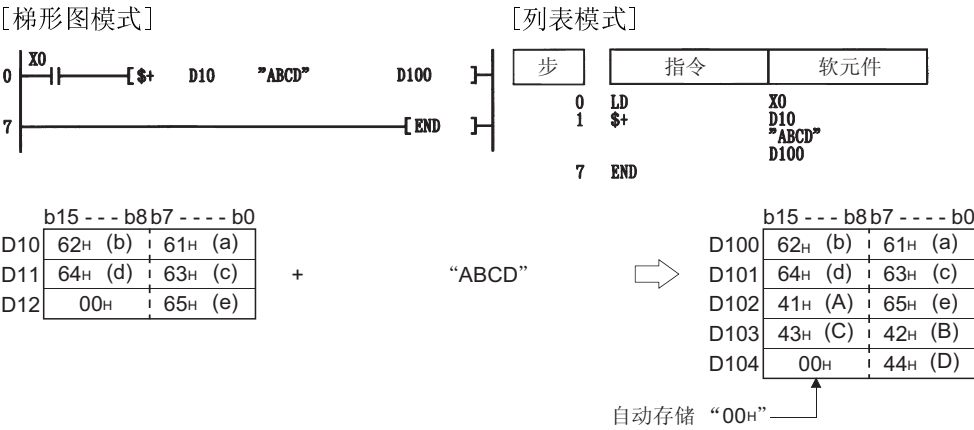
- (2) 当链接字符串的时候，表示Ⓓ指定的字符串数据的结束的“00H”将被忽略，而且将Ⓔ2指定的字符串数据追加到Ⓔ1字符串的最后一个字符之后。

[运行错误]

- (1) 在以下情况中发生运行错误，并且错误标志变为 ON。
- 不能存储从⑤指定的软元件号到相关软元件的最后一个软元件号之间的已完成链接的整个字符串。(错误代码：4101)
 - ⑤或⑥指定的字符串的存储软元件号与⑤指定的字符串的存储软元件号相互重叠。(错误代码：4101)

[程序示例]

- (1) 当 X0 变为 ON 时，以下程序将存储在 D10 到 D12 中的字符串与字符串“ABCD”链接起来，并将它们存储在从 D100 开始的软元件中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.2.13 16 位 BIN 数据的递增和递减运算 (INC, INCP, DEC 与 DECP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
①	○						—		



[设定数据]

设定数据	含义	数据类型
①	执行 INC(加 1)、DEC(减 1)运算的起始软元件号	BIN 16 位

[功能]

INC

(1) 将由①指定的软元件 (16 位数据) 加 1。



(2) 如果①指定的软元件的内容是 32767，并且在该软元件上执行了 INC 或 INCP 指令，那么值-32768 将存储在①指定的软元件中。

DEC

(1) 从①指定的软元件 (16 位数据) 中减 1。



(2) 如果①指定的软元件的内容是 0，并且在该软元件上执行了 DEC 或 DECP 指令，那么值-1 将存储在①指定的软元件中。

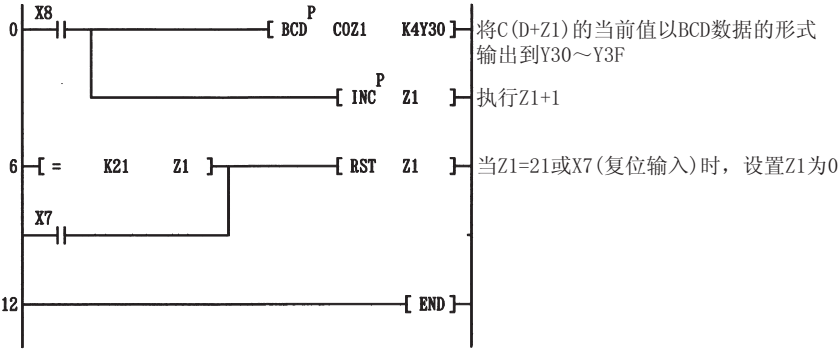
[运行错误]

(1) 没有与 INC, INCP, DEC 或 DECP 指令相关的运行错误。

[程序示例]

(1) 当 X8 变为 ON 时，存储在计数器 C0 到 C20 中的当前值以 BCD 数据的形式输出到 Y30 到 Y3F。
(在当前值小于 9999 的时候)

[梯形图模式]

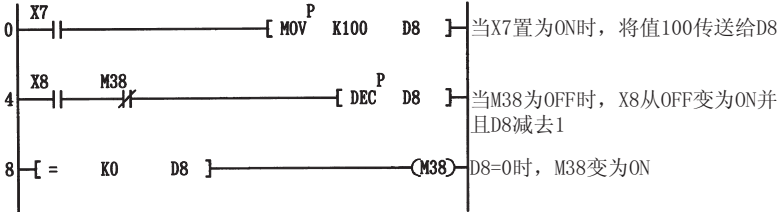


[列表模式]

步	指令	软元件
0	LD	X8
1	BCDP	COZ1 K4Y30
4	INCP	Z1
6	LD=	K21 Z1
9	OR	X7
10	RST	Z1
12	END	

(2) 以下是一个减法计数器程序。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X7
1	MOVP	K100 D8
4	LD	X8
5	ANI	M38
6	DECP	D8
8	LD=	K0 D8
11	OUT	M38
12	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.2.14 32 位 BIN 数据的递增和递减运算(DINC, DINCP, DDEC 与 DDECP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○						—		



[设定数据]

设定数据	含义	数据类型
①	执行 DINC(加 1)、DDEC(减 1) 运算的起始软元件号	BIN 32 位

[功能]

DINC

(1) 将①指定的软元件 (32 位数据) 加 1。



(2) 如果①指定的软元件的内容是 2147483647，并且在该软元件上执行了 DINC 或 DINCP 指令，那么值-2147483648 将存储在①指定的软元件中。

DDEC

(1) 从①指定的软元件 (32 位数据) 中减 1。



(2) 如果①指定的软元件的内容是 0，并且在该软元件上执行了 DDEC 或 DDECP 指令，那么值-1 将存储在①指定的软元件中。

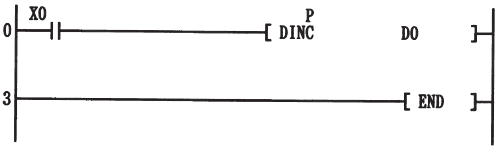
[运行错误]

(1) 不存在与 DINC (P) 或 DDEC (P) 指令相关的运行错误。

[程序示例]

(1) 当 X0 变为 ON 时，以下程序将 D0 到 D1 中的数据加 1。

[梯形图模式]

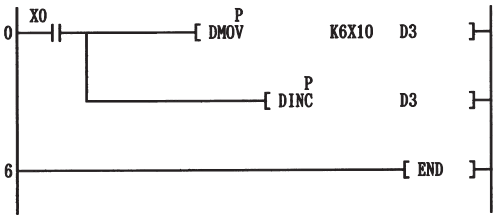


[列表模式]

步	指令	软元件
0	LD	X0
1	DINCP	D0
3	END	

(2) 当 X0 变为 ON 时，以下程序将 X10 到 X27 中的数据加 1，并将其结果存储在 D3 和 D4 中。

[梯形图模式]

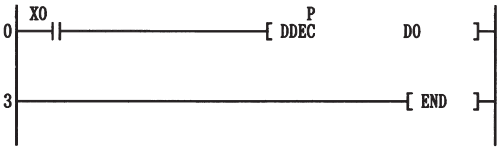


[列表模式]

步	指令	软元件
0	LD	X0
1	DMOVP	K6X10 D3
4	DINCP	D3
6	END	

(3) 当 X0 变为 ON 时，以下程序从 D0 到 D1 中的数据中减 1。

[梯形图模式]

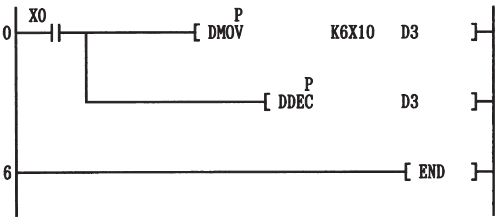


[列表模式]

步	指令	软元件
0	LD	X0
1	DDECP	D0
3	END	

(4) 当 X0 变为 ON 时，以下程序从 X10 到 X27 中的数据中减 1，并将其结果存储在 D3 和 D4 中。

[梯形图模式]



[列表模式]

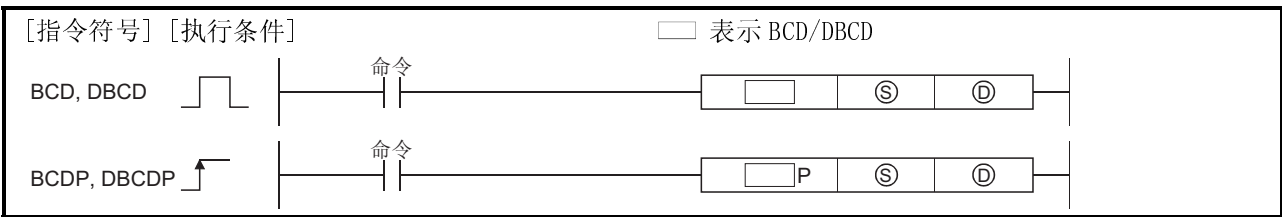
步	指令	软元件
0	LD	X0
1	DMOVP	K6X10 D3
4	DDECP	D3
6	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.3 数据转换指令

6.3.1 从 BIN 数据到 4 位和 8 位 BCD 数据的转换(BCD, BCDP, DBCD 与 DBCDP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]G[][]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	



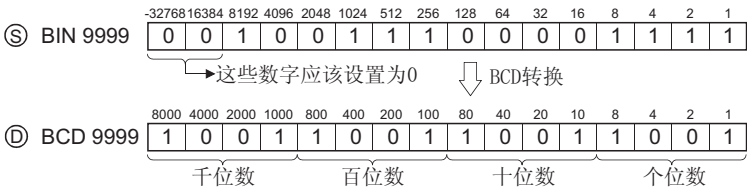
[设定数据]

设定数据	含义	数据类型
⑤	BIN 数据或者存储 BIN 数据的起始软元件号	BIN 16/32 位
⑥	将存储 BCD 数据的起始软元件号	BCD 4/8 位

[功能]

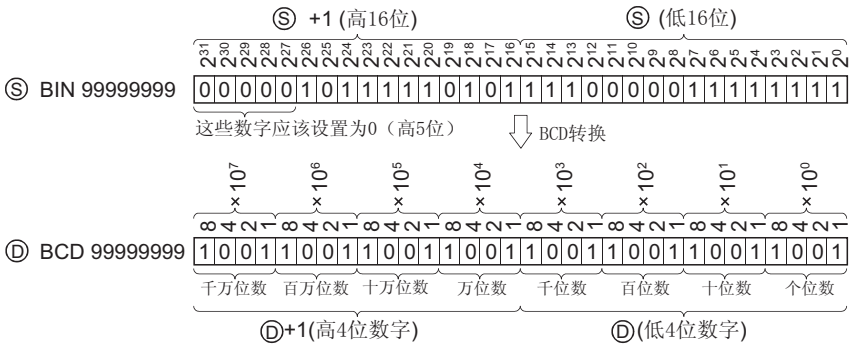
BCD

将⑤指定的软元件中的 BIN 数据 (0 到 9999) 转换成 BCD 数据, 并将它存储在⑥指定的软元件中。



DBCD

将⑤指定的软元件中的 BIN 数据 (0 到 99999999) 转换成 BCD 数据, 并将它存储在⑥指定的软元件中。

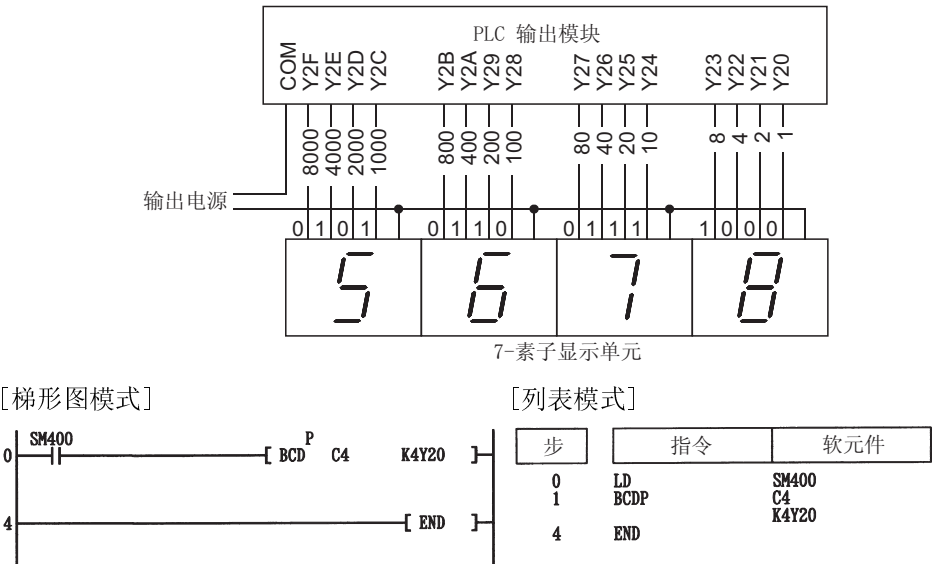


[运行错误]

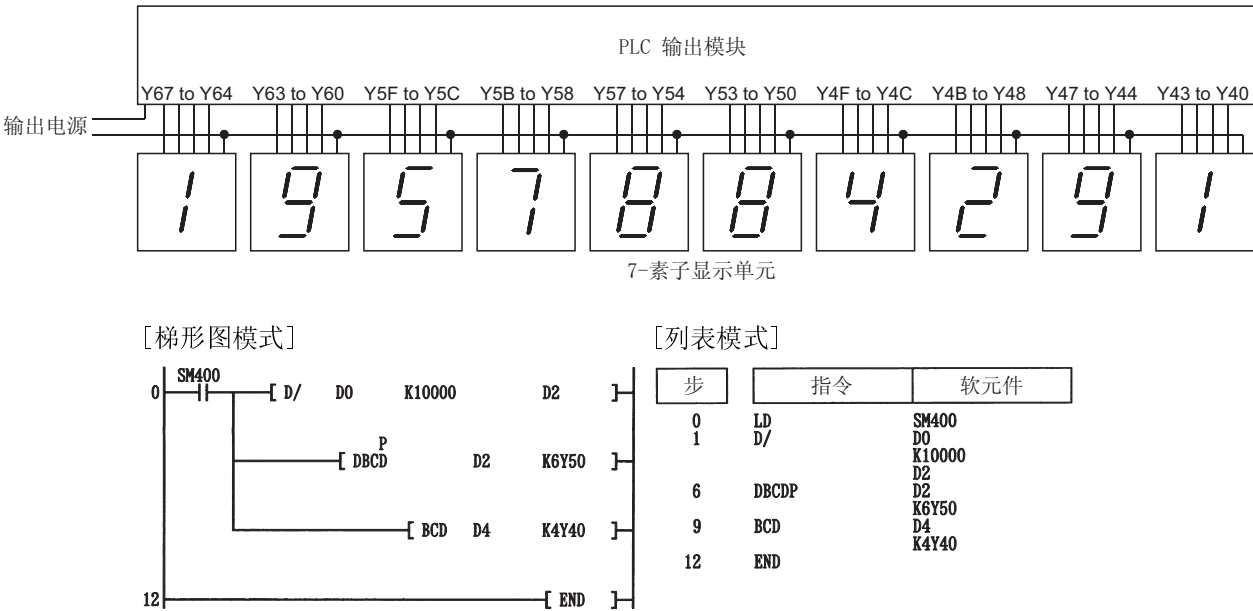
- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 发出 BCD 指令时，⑤上的数据不在 0 到 9999 范围之内。 (错误代码：4100)
 - 发出 DBCD 指令时，⑤和⑤+1 上的数据不在 0 到 99999999 范围之内。 (错误代码：4100)

[程序示例]

- (1) 以下程序将位于 Y20 到 Y2F 上的 C4 的当前值输出到 BCD 显示设备。



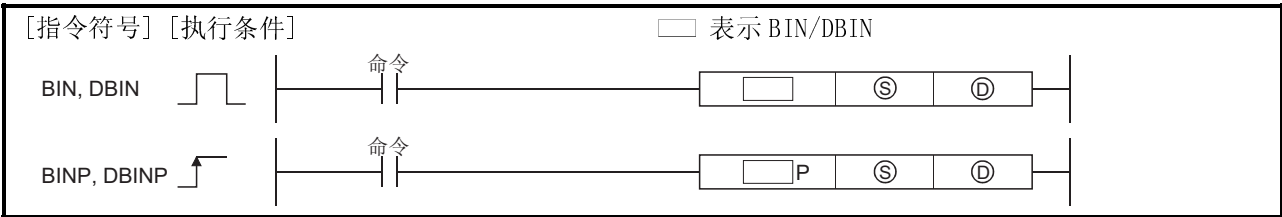
- (2) 以下程序将 D0 和 D1 上的 32 位数据输出到 Y40 到 Y67。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.3.2 从 BCD 4 位和 8 位数据到 BIN 数据的转换 (BIN, BINP, DBIN 与 DBINP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				U
	⑤	○							○
⑥	○					—	—		



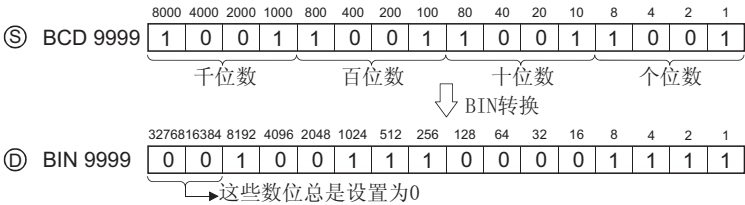
[设定数据]

设定数据	含义	数据类型
⑤	BCD 数据或者存储 BCD 数据的起始软元件号	BCD 4/8 位
⑥	将存储 BIN 数据的起始软元件号	BIN 16/32 位

[功能]

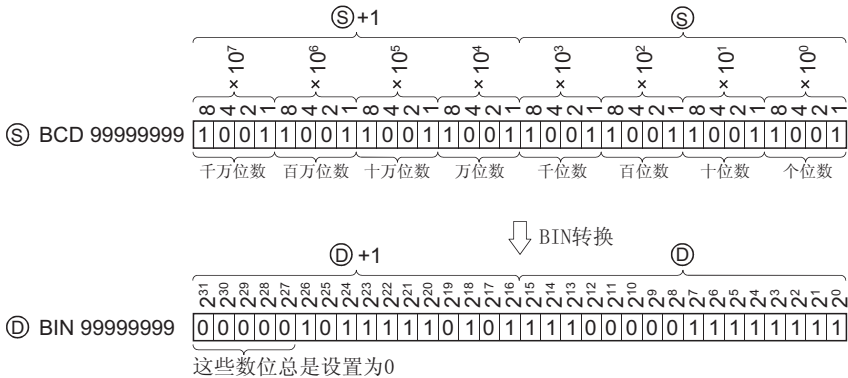
BIN

将⑤指定的软元件中的 BCD 数据 (0 到 9999) 转换成 BIN 数据, 并将它存储在⑥指定的软元件中。



DBIN

将⑤指定的软元件中的 BCD 数据 (0 到 99999999) 转换成 BIN 数据, 并将它存储在⑥指定的软元件中。

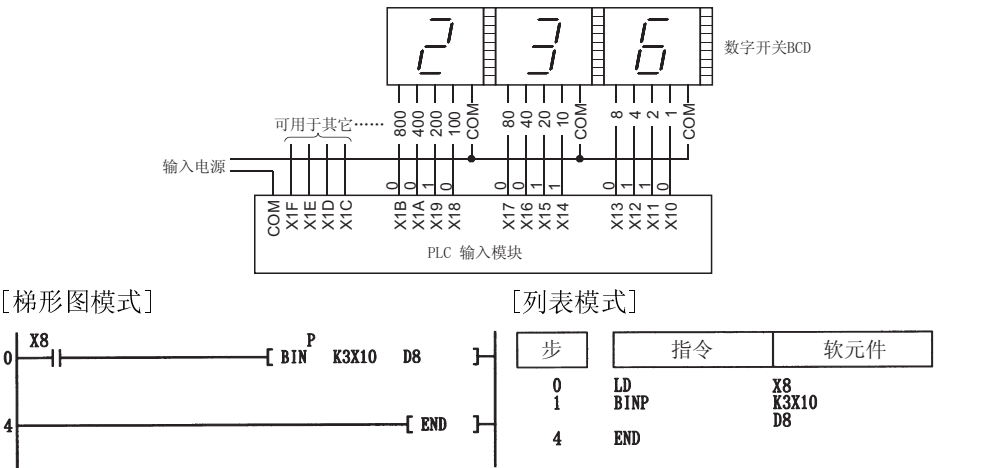


[运行错误]

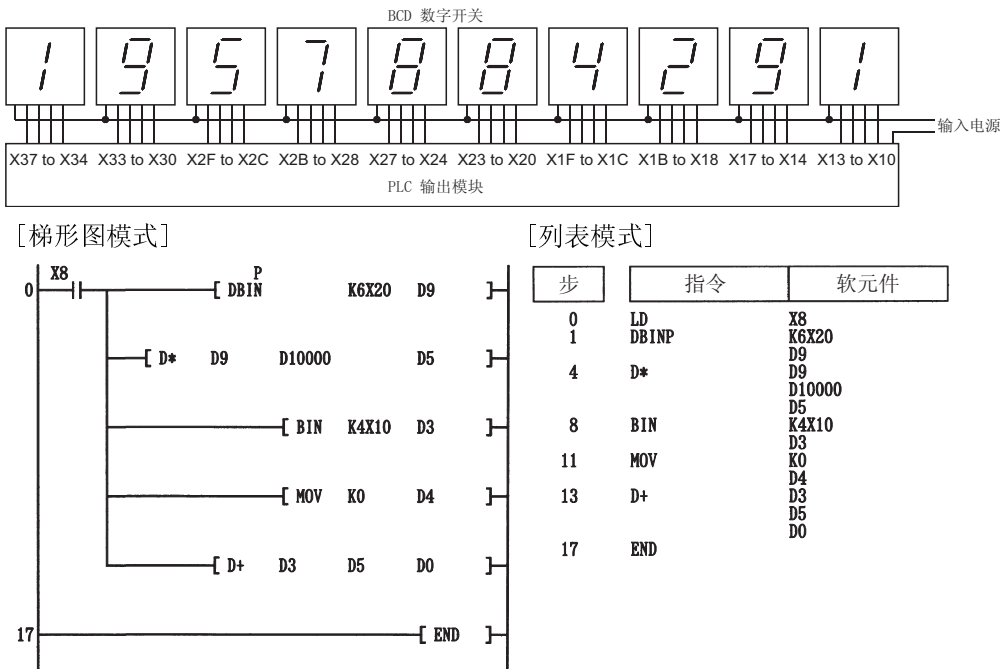
- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中，指令也不会执行。
- 当指定给⑤的任何数字位的值不是 0 到 9 的数字时(错误代码：4100)。
- [当使用 QCPU 的时候]
- 当使用 QCPU 的时候，以上错误可以通过将 SM722 变为 ON 而被抑制掉。
- 然而，如果所指定的值超出了允许的范围，那么不管将 SM722 变为 ON 还是 OFF，指令都不会执行。
- 对 BINP/DBINP 指令来说，不管是否存在错误，下一步运算都要等到命令(执行条件)从 OFF 变为 ON 的时候才执行。

[程序示例]

- (1) 当 X8 变为 ON 时，以下程序将 X10 到 X1B 中的 BCD 数据转换成 BIN 数据，并将它存储在 D8 中。



- (2) 当 X8 变为 ON 时，以下程序将 X10 到 X37 中的 BCD 数据转换成 BIN 数据，并将它存储在 D0 和 D1 中。



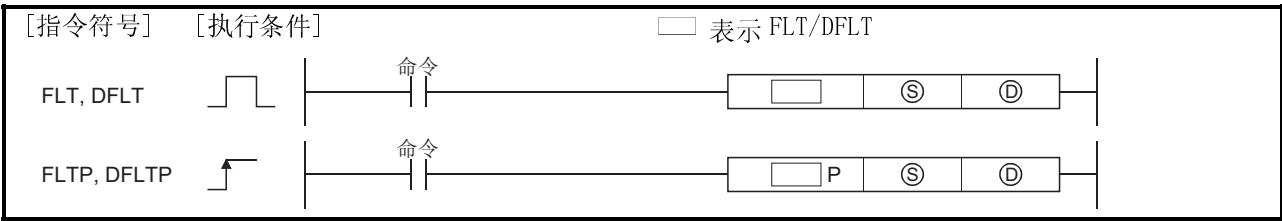
如果指定给 X10 到 X37 的数据是一个 BCD 值而且它大于 2147483647，那么 D0 和 D1 中的值将为负；这是因为它已经超出了 32 位软元件所能处理的数值的范围。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

※: 序列号的前五位数字是04122或者更大的

6.3.3 从 BIN 16 位和 32 位数据到浮点数据的转换 (FLT, FLTP, DFLT 和 DFLTP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]K[]		特殊功能模 块 U[]V[]G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○	○		○	○		○		—
⑥	—	○		—	○		—		—



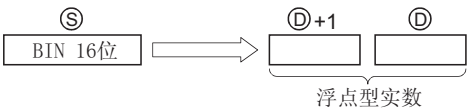
[设定数据]

设定数据	含义	数据类型
Ⓢ	存储将要转换为浮点数据的整数的起始软元件号	BIN 16/32 位
Ⓓ	将用来存储已转换的浮点数据的起始软元件号	实数

[功能]

FLT

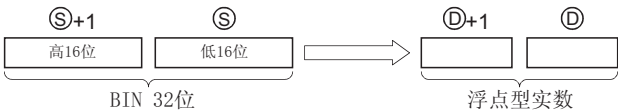
(1) 将Ⓢ指定的 16 位 BIN 数据转换成浮点类型实数，并将其结果存储在Ⓓ指定的软元件号中。



(2) 可以指定给Ⓢ的 BIN 值的范围是-32768 到 32767。

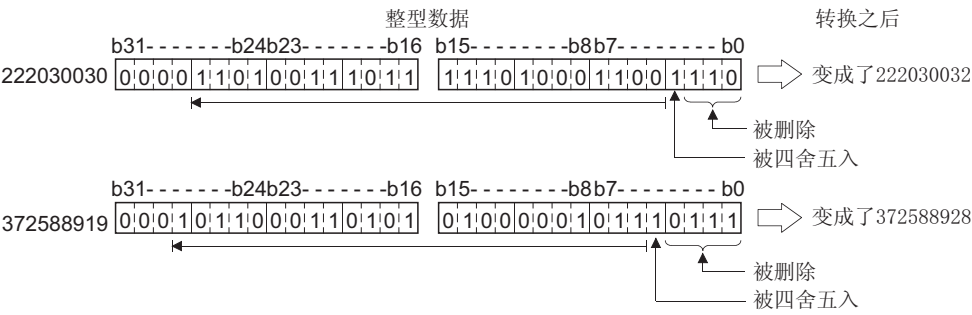
DFLT

(1) 将Ⓢ指定的 32 位 BIN 数据转换成浮点类型实数，并将其结果存储在Ⓓ指定的软元件号中。



(2) 可以指定给Ⓢ+1 和Ⓢ的 BIN 值的范围是-2147483648 到 2147483647。

- (3) 浮动小数点类型实数的处理是通过简单的 32 位处理来实现的。基于这样的事实，如果以二进制形式显示，那么有效数字的位数是 24 位，如果以十进制形式显示，则有效数字的位数大约是 7 位数字。
- 因此，如果该整数超出了-16777216 到 16777215 (24 位 BIN 值) 的范围，则可能在转换的结果值中产生错误。
- 对转换结果从其整型值的最高位开始的第 25 位进行四舍五入，而对其第 26 位及其以后各位中的内容全部予以删除。

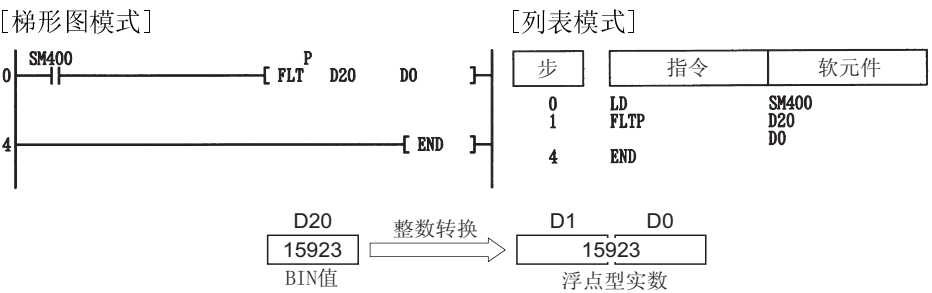


[运行错误]

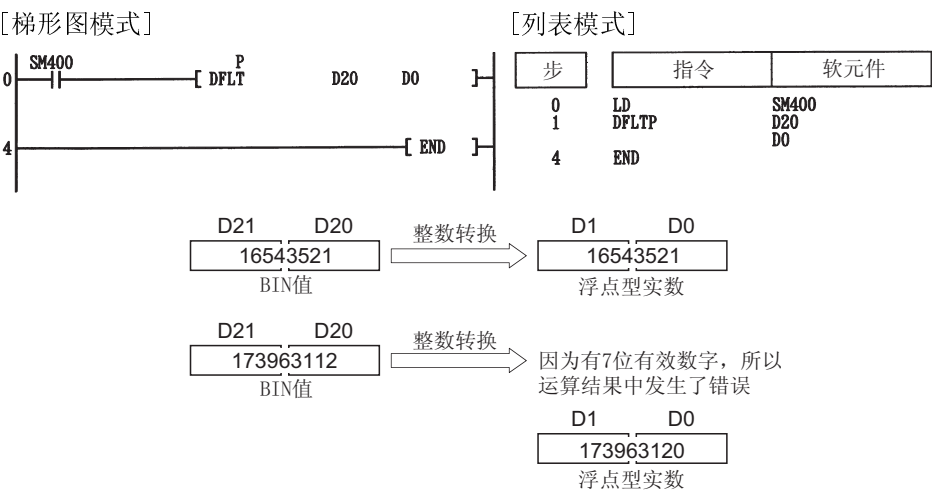
- (1) 不存在与 FLT(P) 或 DFLT(P) 指令相关的运行错误。

[程序示例]

- (1) 以下程序将 D20 上的 BIN 16 位数据转换成浮点类型实数，并将其结果存储在 D0 和 D1 中。



- (2) 以下程序将 D20 和 D21 中的 BIN 16 位数据转换成浮点类型实数，并将其结果存储在 D0 和 D1 中。

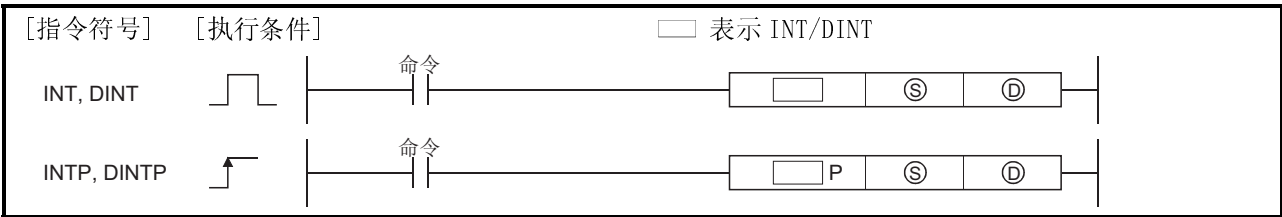


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

※: 序列号的前五位数字是04122或者更大的

6.3.4 从浮点数据到 BIN 16 位和 32 位数据的转换 (INT, INTP, DINT 与 DINTP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J K		特殊功能模 块 U V G	变址寄存器 Zn	常量 E	其它
	位	字		位	字				
⑤	—	○		—	○		—	○	—
⑥	○	○		○	○		○	—	—



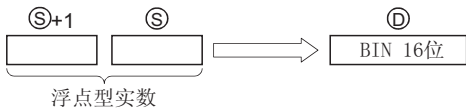
[设定数据]

设定数据	含义	数据类型
⑤	存储将要转换为 BIN 值的浮点数据的起始软元件号	实数
⑥	存储转换后的 BIN 值的起始软元件号	BIN 16/32 位

[功能]

INT

(1) 将⑤指定的浮点类型实数转换成 BIN 16 位数据，并将它存储在⑥指定的软元件号中。



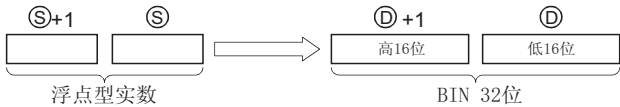
(2) 可以指定给⑤+1 和⑤的浮点类型实数的范围是-32768 到 32767。

(3) 以 BIN 16 位值的形式将整数值存储在⑥中。

(4) 转换后，对该实数小数点后的第一位数字四舍五入。

DINT

(1) 将⑤指定的浮点类型实数转换成 BIN 32 位数据，并将其结果存储在⑥指定的软元件号中。



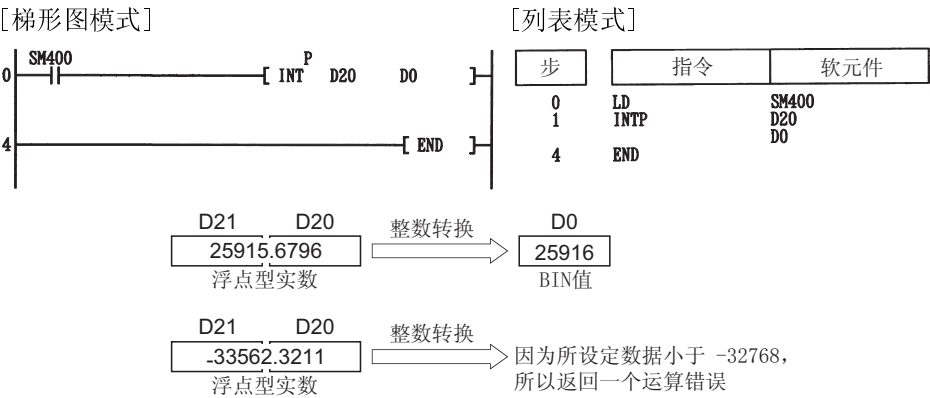
- (2) 可以指定给⑤+1 和⑤的浮点类型实数的范围是-2147483648 到 2147483647。
- (3) 以 BIN 32 位值的形式将整数值存储在⑤+1 和⑤中。
- (4) 转换后，对该实数小数点后的第一位数字四舍五入。

[运行错误]

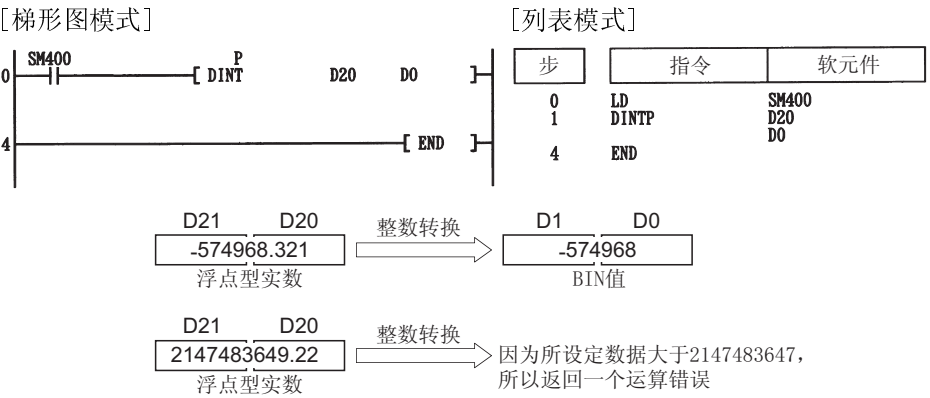
- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 使用 INT 指令的时候，指定给⑤的浮点类型数据超出了-31768 到 32767 范围。
 - 使用 DINT 指令的时候，指定给⑤的浮点类型数据超出了-2147483648 到 2147483647 范围。

[程序示例]

- (1) 以下程序将 D20 和 D21 中的浮点类型实数转换成 BIN 16 位数据，并将其结果存储在 D0 中。



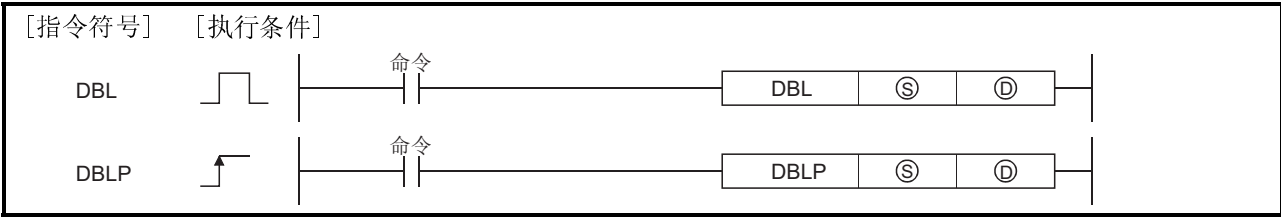
- (2) 以下程序将 D20 和 D21 中的浮点类型实数转换成 BIN 32 位数据，并将其结果存储在 D0 和 D1 中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.3.5 从 BIN 16 位数据到 BIN 32 位数据的转换 (DBL 与 DBLP)

设定数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	

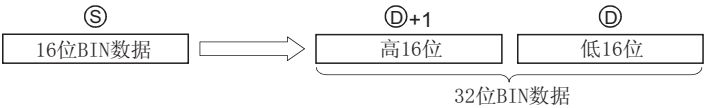


[设定数据]

设定数据	含义	数据类型
Ⓢ	存储 BIN 16 位数据的起始软元件号	BIN 16 位
ⓓ	转换完成后, 存储 BIN 32 位数据的起始软元件号	BIN 32 位

[功能]

将Ⓢ指定的软元件中的 BIN 16 位转换为带符号的 BIN 32 位数据, 并将其结果存储在ⓓ指定的软元件中。

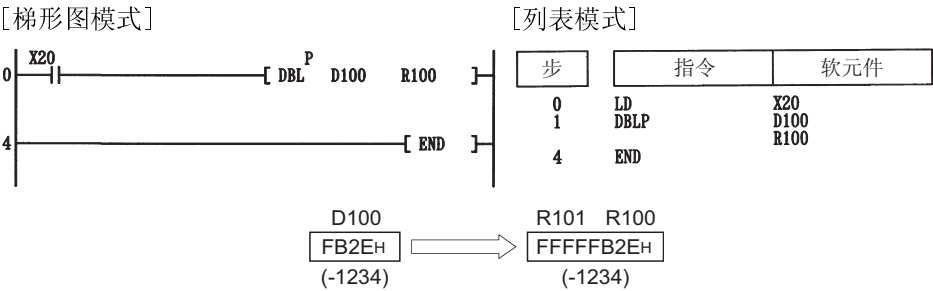


[运行错误]

(1) 不存在与 DBL (P) 指令相关的运行错误。

[程序示例]

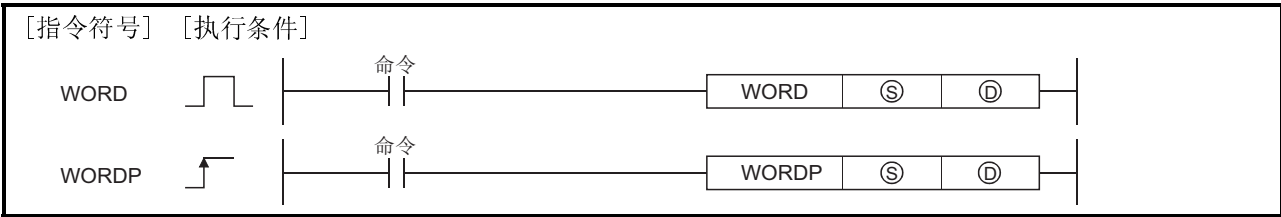
(1) 当 X20 变为 ON 时, 以下程序将存储在 D100 中的 BIN 16 位数据转换成 BIN 32 位数据, 并将它存储在 R100 和 R101 中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.3.6 从 BIN 32 位到 BIN 16 位数据的转换(WORD 与 WORDP)

设定数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	

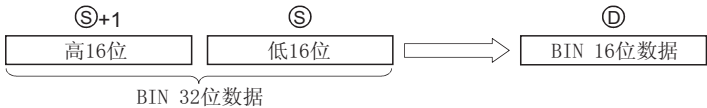


[设定数据]

设定数据	含义	数据类型
⑤	存储 BIN 32 位数据的起始软元件号	BIN 32 位
⑥	转换完成后, 存储 BIN 16 位数据的起始软元件号	BIN 16 位

[功能]

将⑤指定的软元件中的 BIN 32 位转换为带符号的 BIN 16 位数据, 并将其结果存储在⑥指定的软元件中。
可指定给软元件的值的范围是-32768 到 32767。



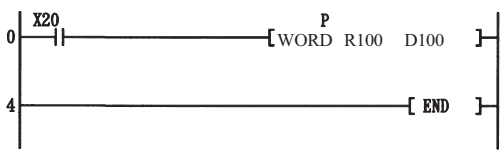
[运行错误]

- (1) 在发生以下运行错误的情况中, 错误标志(SM0)变为 ON, 并且错误代码存储在 SD0 中。
- 指定给⑤+1 和⑤的数据的值不在-31768 到 32767 范围之内。(错误代码: 4100)

[程序示例]

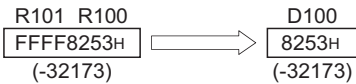
- (1) 当 X20 变为 ON 时, 以下程序将存储在 R100 和 R101 中的 BIN 32 位数据转换成 BIN 16 位数据, 并将它存储在 D100 中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X20
1	WORD P	R100 D100
4	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.3.7 从 BIN 16 位和 32 位数据到格雷码的转换 (GRY, GRYP, DGRY 与 DGRYP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	

[指令符号]	[执行条件]	□ 表示 GRY/DGRY	
GRY, DGRY			⑤ ⑥
GRYP, DGRYP			⑤ ⑥

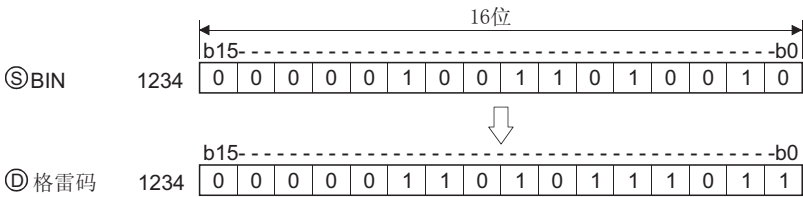
[设定数据]

设定数据	含义	数据类型
⑤	BIN 数据或存储 BIN 数据的起始软元件号	BIN 16/32 位
⑥	转换完成后, 存储格雷码的起始软元件号	BIN 16/32 位

[功能]

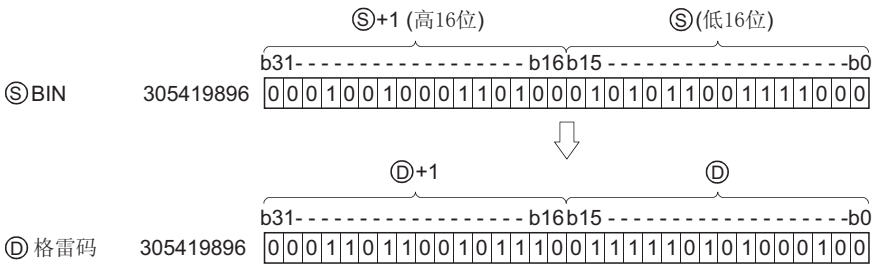
GRY

将⑤指定的软元件中的 BIN 16 位数据转换为格雷码, 并将其结果存储在⑥指定的软元件中。



DGRY

将⑤指定的软元件中的 BIN 32 位数据转换为格雷码, 并将其结果存储在⑥指定的软元件中。



[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- ⑤中的数据是负数。(错误代码：4100)

[程序示例]

- (1) 当 X10 变为 ON 时，以下程序将 D100 中的 BIN 数据转换成格雷码，并将其结果存储在 D200 中。

[梯形图模式]

```
graph LR
    subgraph Step0 [0]
        X10[X10] --> PGRY[P GRY]
        PGRY --> D100[D100]
        D100 --> D200[D200]
    end
    subgraph Step4 [4]
        END[END]
    end
```

[列表模式]

步	指令	软元件
0	LD	X10
1	GRYP	D100
		D200
4	END	

- (2) 当 X1C 变为 ON 时，以下程序将 D10 和 D11 中的 BIN 数据转换成格雷码，并将它存储在 D100 和 D101 中。

[梯形图模式]

```
graph LR
    subgraph Step0 [0]
        X1C[X1C] --> PDGRY[P DGRY]
        PDGRY --> D10[D10]
        D10 --> D100[D100]
    end
    subgraph Step4 [4]
        END[END]
    end
```

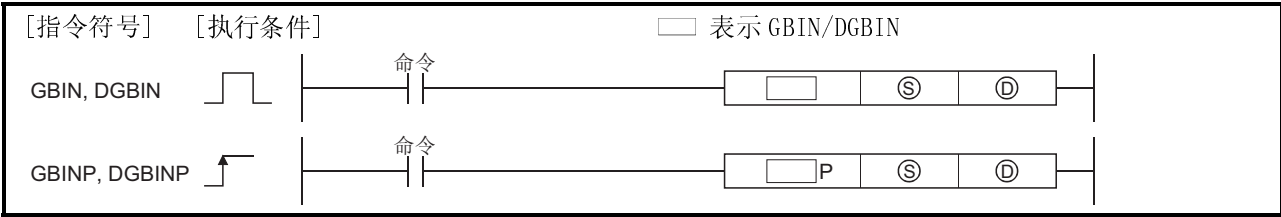
[列表模式]

步	指令	软元件
0	LD	X1C
1	DGRYP	D10
		D100
4	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.3.8 从格雷码到 BIN 16 位和 32 位数据的转换 (GBIN, GBINP, DGBIN 与 DGBINP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	



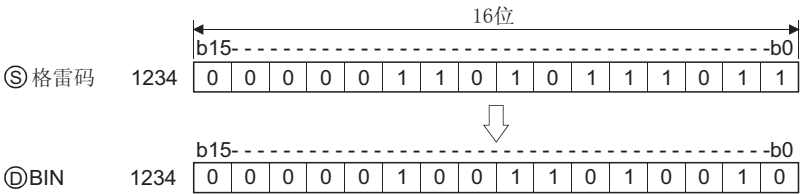
[设定数据]

设定数据	含义	数据类型
⑤	格雷码或存储格雷码的起始软元件号	BIN 16/32 位
⑥	转换完成后, 存储 BIN 数据的起始软元件号	BIN 16/32 位

[功能]

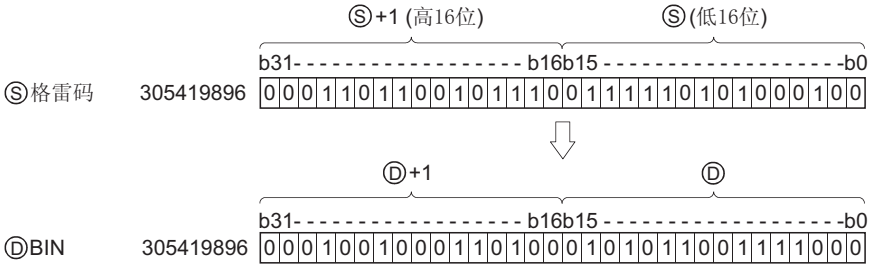
GBIN

将⑤指定的软元件中的格雷码数据转换成 BIN 16 位数据, 并存储在⑥指定的软元件中。



DGBIN

将⑤指定的软元件中的格雷码数据转换成 BIN 32 位数据, 并存储在⑥指定的软元件中。

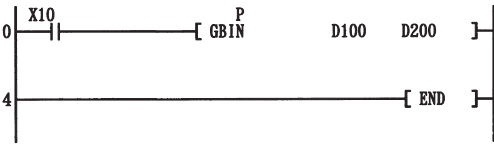


[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 发出 GBIN 指令时，⑤中的数据不在 0 到 32767 范围之内。(错误代码：4100)
 - 发出 DGBIN 指令时，⑤中的数据不在 0 到 2147483647 范围之内。(错误代码：4100)

[程序示例]

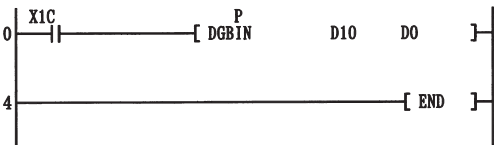
- (1) 当 X10 变为 ON 时，以下程序将 D100 中的格雷码数据转换成 BIN 数据，并将其结果存储在 D200 中。

[梯形图模式]

[列表模式]

步	指令	软元件
0	LD	X10
1	GBINP	D100 D200
4	END	

- (2) 当 X1C 变为 ON 时，以下程序将 D10 和 D11 中的格雷码数据转换成 BIN 数据，并将其结果存储在 D0 和 D1 中。

[梯形图模式]

[列表模式]

步	指令	软元件
0	LD	X1C
1	DGBINP	D10 D0
4	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.3.9 BIN 16 位和 32 位数据的 2 进制补码(符号取反) (NEG, NEGP, DNEG 与 DNEGP)

设定 数据	可用软元件								
	内部软元件 (系统、用户)		文件寄存器	MELSECNET/10 (H) 直接 J□□□□		特殊功能模 块 U□□□□	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
①	○						—		



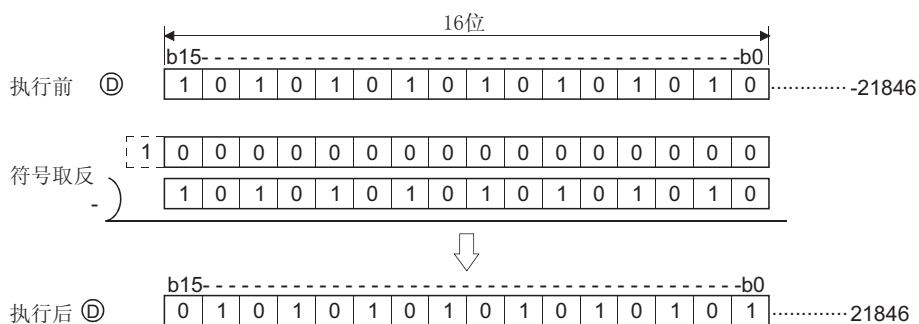
[设定数据]

设定数据	含义	数据类型
⑤	存储用于二进制补码操作的数据的起始软元件号	BIN 16/32 位

〔功能〕

NEG

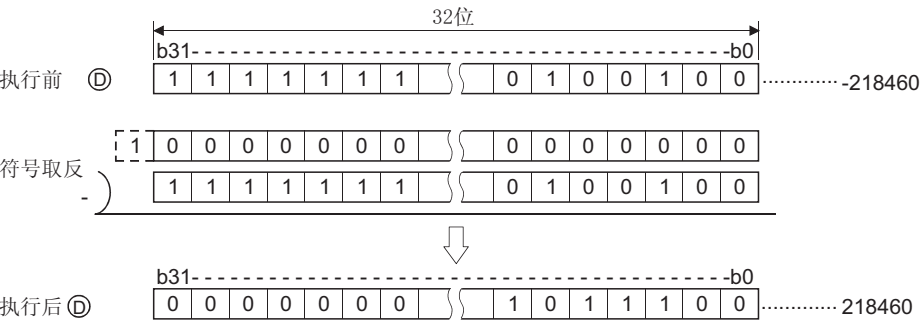
(1) 将⑤指定的软元件中的 16 位数据的符号取反, 并将其结果存储在⑥指定的软元件中。



(2) 用于对正号和负号取反。

DNEG

(1) 将Ⓓ指定的软元件中的 32 位数据的符号取反，并将其结果存储在Ⓓ指定的软元件中。



(2) 用于对正号和负号取反。

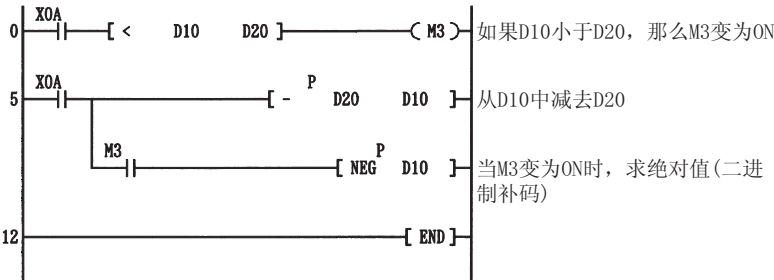
[运行错误]

(1) 不存在与 NEG (P) 或 DNEG (P) 指令相关的运行错误。

[程序示例]

(1) 当 XA 变为 ON 时，以下程序计算 D10 到 D20 中的数据总和，如果其结果为负数，那么求它的绝对值。

[梯形图模式]



[列表模式]

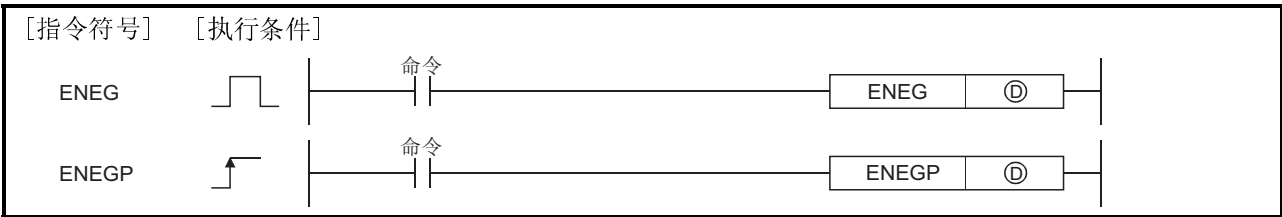
步	指令	软元件
0	LD	X0A
1	AND<	D10 D20
4	OUT	M3
5	LD	X0A
6	-P	D20 D10
8	AND	M3
9	NEGP	D10
12	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

※: 序列号的前五位数字是04122或者更大的

6.3.10 浮点数据的符号取反(ENEG 与 ENEGP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	—	○		—	○		—	—	



[设定数据]

设定数据	含义	数据类型
Ⓓ	存储符号将要取反的浮点数据的起始软元件号	实数

[功能]

- (1) 将Ⓓ指定的浮点类型实数的符号取反，并将其结果存储在Ⓓ指定的软元件中。
- (2) 用于对正号和负号取反。

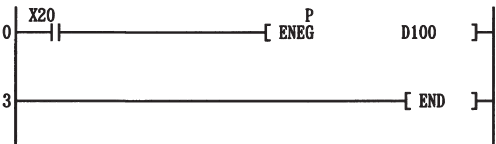
[运行错误]

- (1) 不存在与 ENEG (P) 指令相关的运行错误。

[程序示例]

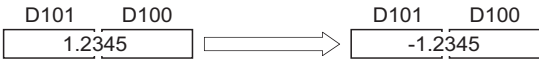
- (1) 当 X20 变为 ON 时，以下程序对 D100 到 D101 中的浮动小数点类型实数的符号取反，并将其结果存储在 D100 到 D101 中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X20
1	ENEGP	D100
3	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.3.11 从块 BIN 16 位数据到 BCD 4 位数据的转换(BKBCD 与 BKBCDP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		常量 K, H
⑤	—	○				—	—
⑥	—	○				—	—
n	○	○				○	—

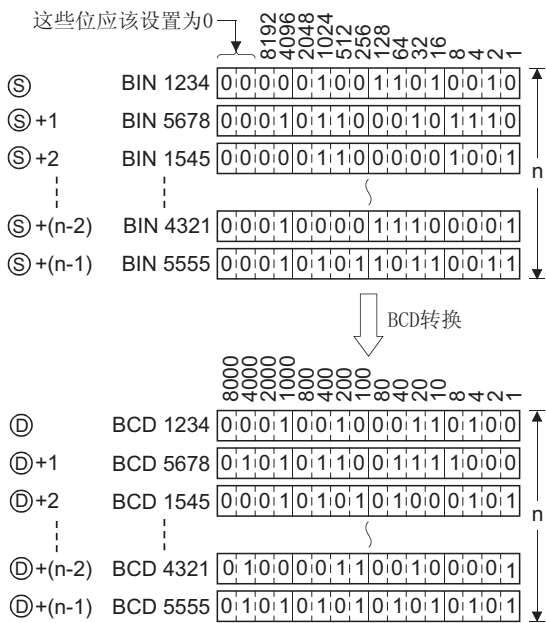
[指令符号]	[执行条件]
BKBCD	
BKBCDP	

[设定数据]

设定数据	含义	数据类型
⑤	存储 BIN 数据的起始软元件号	BIN 16 位
⑥	转换完成后, 将用来存储 BCD 数据的起始软元件号	BCD 4 位
n	被转换数据块的数量	BIN 16 位

[功能]

- (1) 将从⑤指定的软元件开始的 n 点 BIN 数据(0 到 9999)转换成 BCD 数据, 并将其结果存储在从⑥指定的软元件开始的软元件中。



[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 从⑤或⑥软元件开始的 n 点范围超出了相关软元件的范围。

(错误代码：4101)

● 从⑤指定的软元件开始的 n 点数据不在 0 到 9999 范围之内。

(错误代码：4100)

● ⑤软元件与⑥软元件相互重叠。

(错误代码：4101)
- (2) 有关变址修改过程中的错误的信息，请参阅 3.6 节。

[程序示例]

- (1) 当 X20 变为 ON 时，以下程序将起始于 D100 的软元件中的若干个(其个数由存储在 D0 中的值决定)BIN 16 位数据转换为 BCD 数据，并将其结果存储在起始于 D200 的软元件中。

[梯形图模式]

0

X20

P

[BKBCD

D100

D200

D0

]

5

[END

]

[列表模式]

步	指令	软元件
0	LD	X20
1	BKBCDP	D100 D200 D0
5	END	

8192
4096
2048
1024
512
256
128
64
32
16
8
4
2
1

D100 BIN 5432

0 0 0 1 0 1 0 1 0 0 1 1 1 0 0 0

D101 BIN 4444

0 0 0 1 0 0 0 1 0 1 0 1 1 1 0 0

D102 BIN 3210

0 0 0 0 1 1 0 0 1 0 0 0 1 0 1 0

BCD转换

D0 3

8000
4000
2000
1000
800
400
200
100
80
40
20
10
8
4
2
1

D200 BCD 5432

0 1 0 1 0 1 0 0 0 0 1 1 0 0 1 0

D201 BCD 4444

0 1 0 0 0 1 0 0 0 0 1 0 0 0 1 0 0

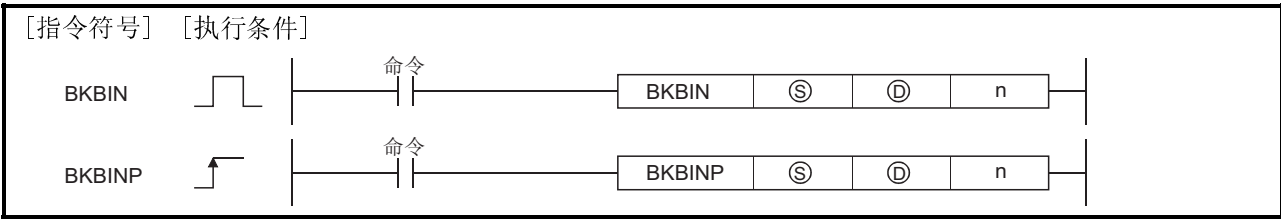
D202 BCD 3210

0 0 1 1 0 0 1 0 0 0 0 1 0 0 0 0

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.3.12 从块 BCD 4 位数据到块 BIN 16 位数据的转换(BKBIN 与 BKBINP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		变址寄存器 Zn	常量 K, H
	位	字		位	字		其它
Ⓔ	—	○			—		—
Ⓕ	—	○			—		—
n	○	○			○		—

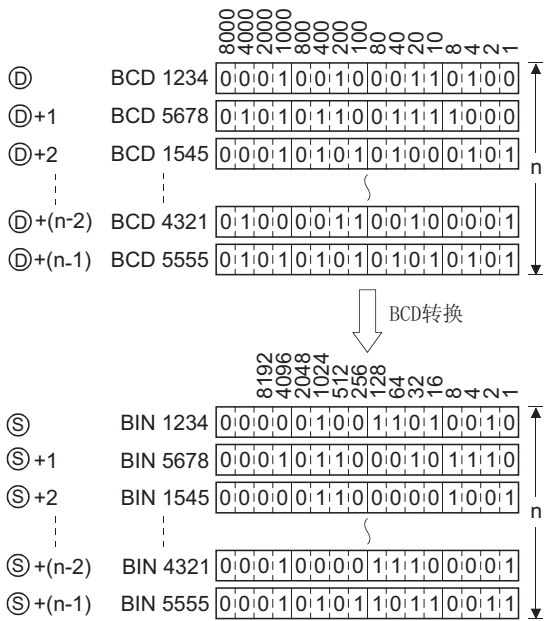


[设定数据]

设定数据	含义	数据类型
Ⓔ	存储 BCD 数据的起始软元件号	BCD 4 位
Ⓕ	转换完成后, 将用来存储 BIN 数据的起始软元件号	BIN 16 位
n	被转换数据块的数量	

[功能]

(1) 将从Ⓔ指定的软元件开始的 n 点 BCD 数据(0 到 9999)转换成 BIN 数据, 并将其结果存储在从Ⓕ指定的软元件开始的软元件中。



[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 从⑤或⑥软元件开始的 n 点范围超出了该软元件本身的范围。

(错误代码: 4101)

● 从⑤软元件开始的 n 点数据不在 0 到 9999 范围之内。

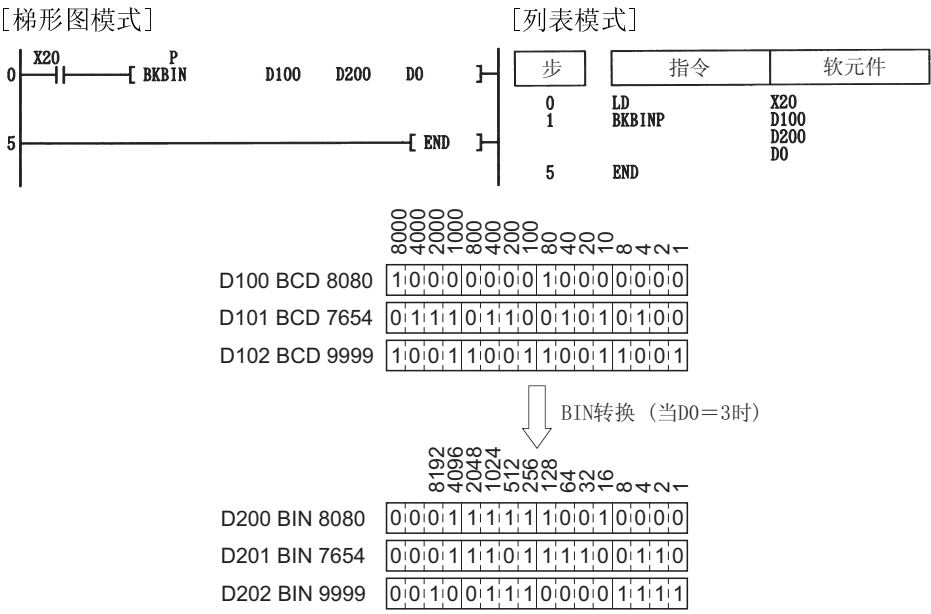
(错误代码: 4100)

● ⑤软元件与⑥软元件相互重叠。

(错误代码: 4101)

[程序示例]

- (1) 当 X20 变为 ON 时，以下程序将起始于 D100 的软元件中的若干个(其个数由存储在 D0 中的值决定)BCD 数据转换为 BIN 数据，并将其结果存储在起始于 D200 的软元件中。

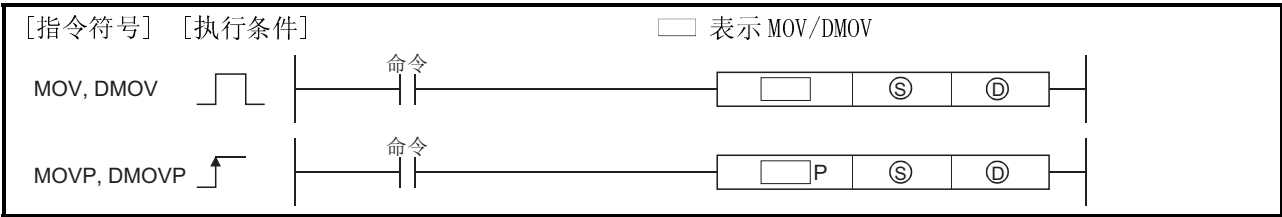


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.4 数据传送指令

6.4.1 16 位和 32 位数据传送 (MOV, MOVP, DMOV 与 DMOVP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][]G[][]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	



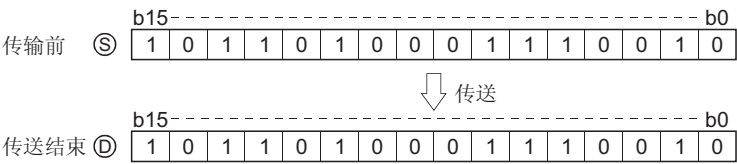
[设定数据]

设定数据	含义	数据类型
⑤	传送数据, 或存储传送数据的软元件号	BIN 16/32 位
⑥	存储已传送数据的软元件号	

[功能]

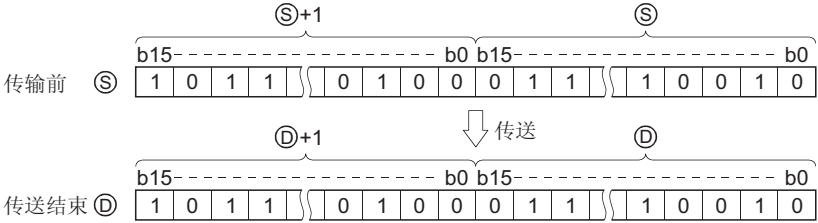
MOV

(1) 将⑤指定的软元件中的 16 位数据传送到⑥指定的软元件。



DMOV

(2) 将⑤指定的软元件中的 32 位数据传送到⑥指定的软元件。



[运行错误]

(1) 不存在与 MOV (P) 或 DMOV (P) 指令相关的运行错误。

[程序示例]

(1) 以下程序将来自 X0 到 XB 的输入数据存储在 D8 中。

[梯形图模式]

[列表模式]

步	指令	软元件
0	LD	SM400
1	MOVP	K3X0
		D8
4	END	

(2) 当 X8 变为 ON 时，以下程序将常数 K155 存储在 D8 中。

[梯形图模式]

[列表模式]

步	指令	软元件
0	LD	X8
1	MOVP	K155
		D8
4	END	

(3) 以下程序将来自 D0 和 D1 的数据存储在 D7 和 D8 中。

[梯形图模式]

[列表模式]

步	指令	软元件
0	LD	SM400
1	DMOVP	D0
		D7
4	END	

(4) 以下程序将来自 X0 和 X1F 的数据存储在 D0 和 D1 中。

[梯形图模式]

[列表模式]

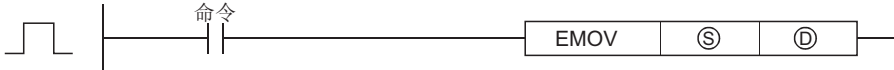
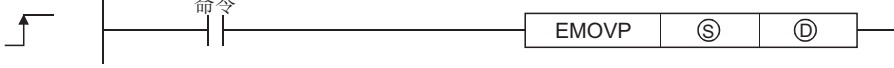
步	指令	软元件
0	LD	SM400
1	DMOVP	K8X0
		D0
4	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

※: 序列号的前五位数字是04122或者更大的

6. 4. 2 浮点数据传送(EMOV 与 EMOVP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○		—	○		—
⑥	—	○		—	○		—

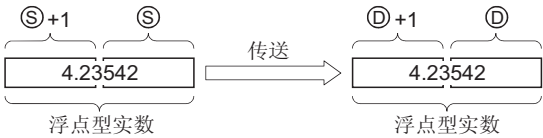
[指令符号]	[执行条件]
EMOV	
EMOVP	

[设定数据]

设定数据	含义	数据类型
⑤	传送数据, 或存储传送数据的软元件号	实数
⑥	存储已传送数据的软元件的号	

[功能]

(1) 将⑤指定的软元件中存储的浮点类型实数传送到⑥指定的软元件中。

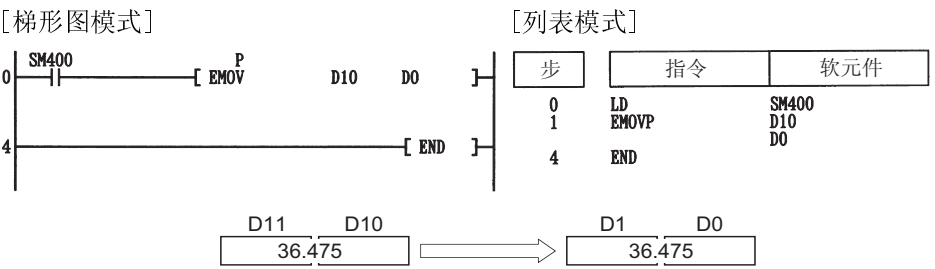


[运行错误]

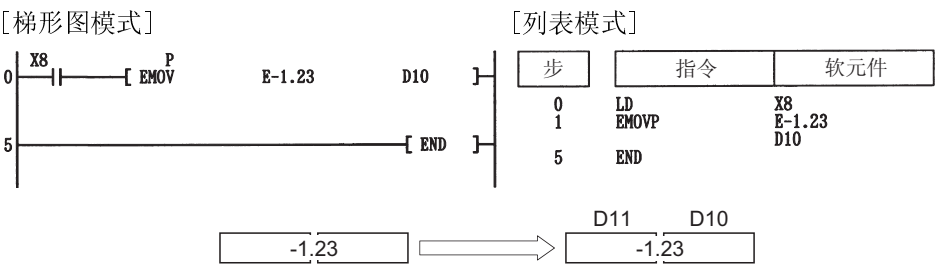
(1) 不存在与 EMOV (P) 指令相关的运行错误。

[程序示例]

(1) 以下程序将 D10 和 D11 中的实数存储在 D0 和 D1 中。



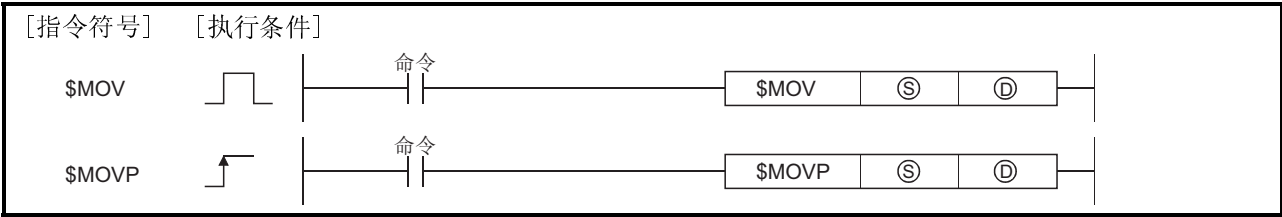
(2) 当 X8 变为 ON 时，以下程序将实数-1.23 存储在 D10 和 D11 中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.4.3 字符串传送(\$MOV 与\$MOVP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		变址寄存器 Zn	常量 \$
	位	字		位	字		其它
⑤	—	○		—			○
⑥	—	○		—			—

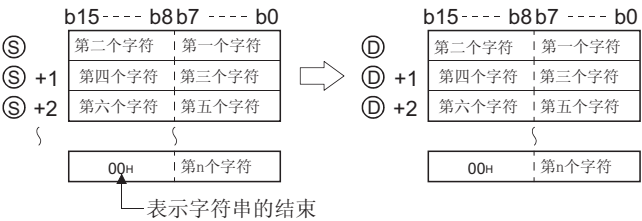


[设定数据]

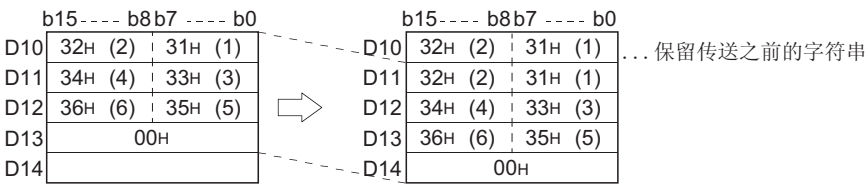
设定数据	含义	数据类型
⑤	将传送的字符串(字符串中的最多字符数: 对于 QnA/Q4AR, 16 个字符; 对于 QCPU, 32 个字符), 或者存储字符串的起始软元件号	字符串
⑥	存储已传送字符串的起始软元件号	

[功能]

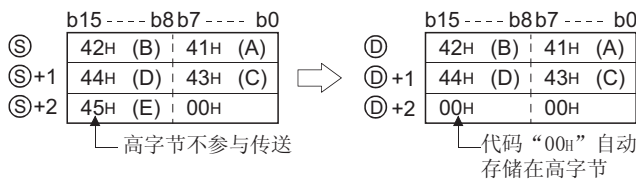
- (1) 将从⑤指定的软元件号开始的软元件中存储的字符串数据传送到从⑥指定的软元件号开始的软元件中。
- 字符串传送包括对从⑤指定的软元件号起始到存储“00H”的软元件号为止的软元件号中存储的所有数据的传送。



(2) 即使遇到存储将要传送的字符数据的软元件范围 (S 到 S+n) 与将要存储已完成传输的字符串数据的软元件范围 (D 到 D+n) 相互重叠的情况，执行处理时也不会发生错误。
当把存储在 D10 到 D13 中的字符串数据传送到 D11 到 D14 中时，会出现以下情况：



(3) 如果代码 “00H” 存储在 S+n 的低字节，那么它会同时存储在 D+n 的高字节和低字节。

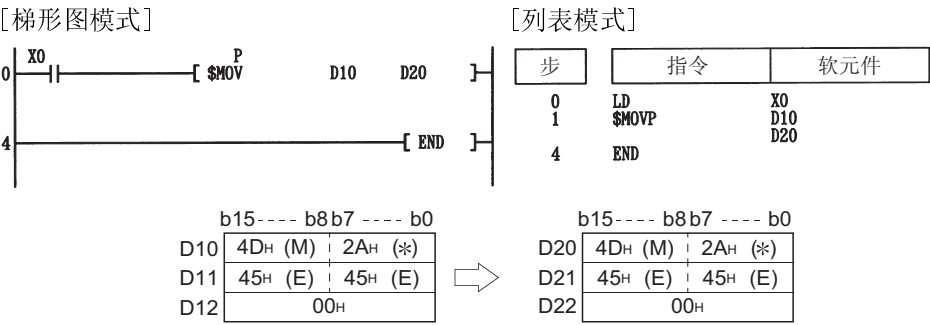


[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 在 S 指定的软元件号与相关软元件之间没有存储代码 “00H”。 (错误代码：4101)
 - 从 D 指定的软元件到引用的最后一个软元件号之间的点数没有足够空间存储整个字符串。 (错误代码：4101)

[程序示例]

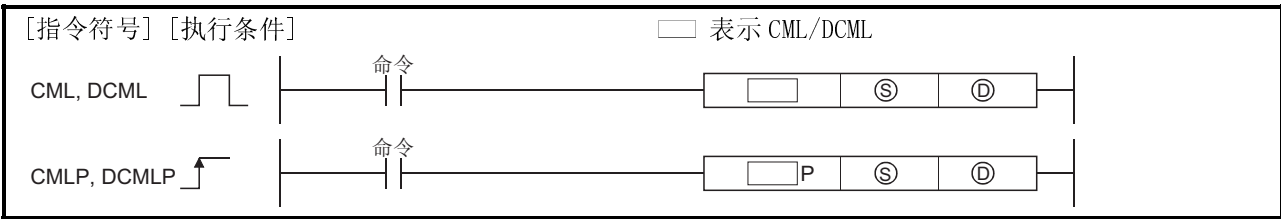
(1) 当 X0 变为 ON 时，将存储在 D10 和 D12 中的字符串数据传送到 D20GCG 到 D22 中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.4.4 16 位和 32 位数据取反传送 (CML, CMLP, DCML 与 DCMLP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	



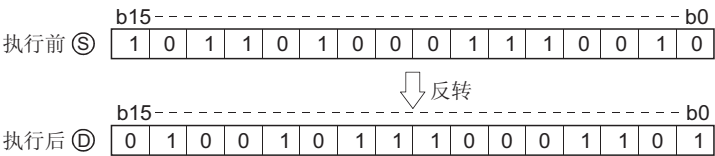
[设定数据]

设定数据	含义	数据类型
Ⓔ	将要取反的数据或者存储该数据的软元件号	BIN 16/32 位
Ⓕ	存储取反结果的软元件号	

[功能]

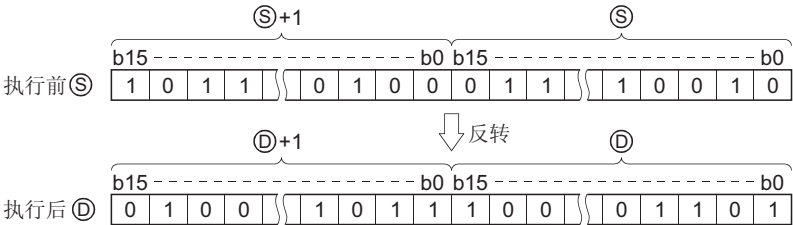
CML

(1) 将Ⓔ指定的 16 位数据逐位取反，并将其结果传送到Ⓕ指定的软元件中。



DCML

(1) 将Ⓔ指定的 32 位数据逐位取反，并将其结果传送到Ⓕ指定的软元件中。



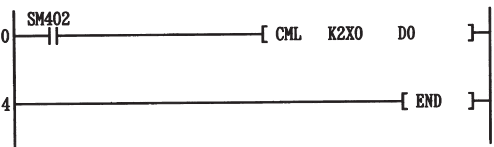
[运行错误]

(1) 不存在与 CML (P) 或 DCML (P) 指令相关的运行错误。

[程序示例]

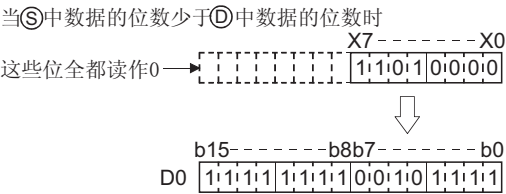
(1) 以下程序将 X0 到 X7 中的数据取反，并将其结果传送到 D0 中。

[梯形图模式]



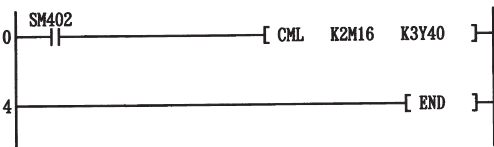
[列表模式]

步	指令	软元件
0	LD	SM402
1	CML	K2X0
4	END	D0



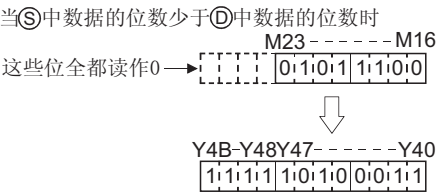
(2) 以下程序将 M16 到 M23 中的数据取反，并将其结果传送到 Y40 到 Y47 中。

[梯形图模式]



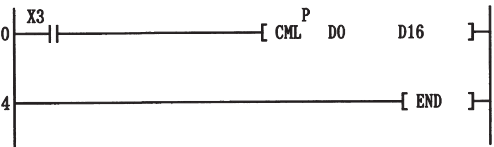
[列表模式]

步	指令	软元件
0	LD	SM402
1	CML	K2M16
4	END	K3Y40



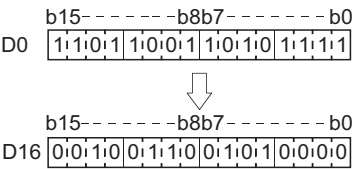
(3) 当 X3 变为 ON 时，以下程序将 D0 中的数据取反，并将其结果传送到 D16 中。

[梯形图模式]



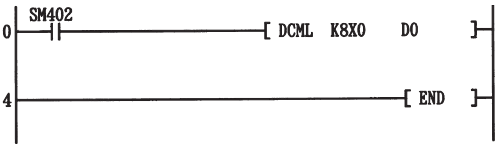
[列表模式]

步	指令	软元件
0	LD	X3
1	CMLP	D0
4	END	D16



(4) 以下程序将 X0 到 X1F 中的数据取反，并将其结果传送到 D0 和 D1 中。

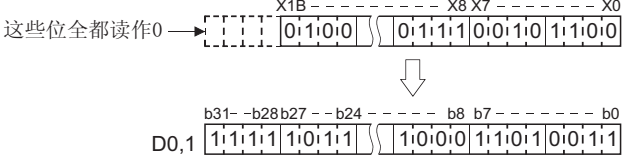
[梯形图模式]



[列表模式]

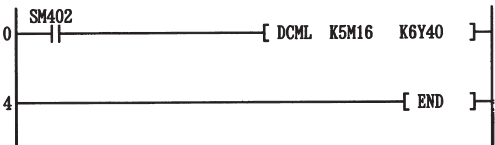
步	指令	软元件
0	LD	SM402
1	DCML	K8X0
		D0
4	END	

当⑤中数据的位数少于④中数据的位数时



(5) 以下程序将 M16 到 M35 中的数据取反，并将它传送到 Y40 到 Y63 中。

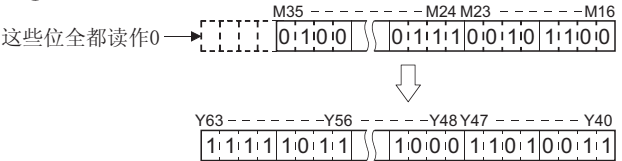
[梯形图模式]



[列表模式]

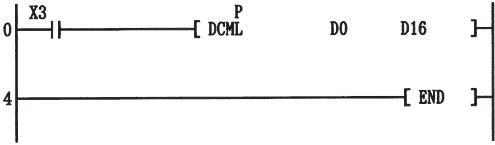
步	指令	软元件
0	LD	SM402
1	DCML	K5M16
		K6Y40
4	END	

当⑤中数据的位数少于④中数据的位数时



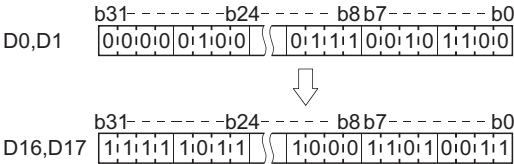
(6) 当 X3 变为 ON 时，以下程序将 D0 和 D1 中的数据取反，并将其结果存储在 D16 和 D17 中。

[梯形图模式]



[列表模式]

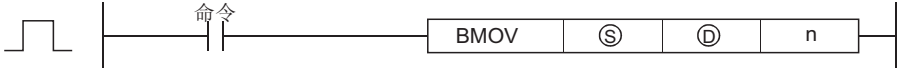
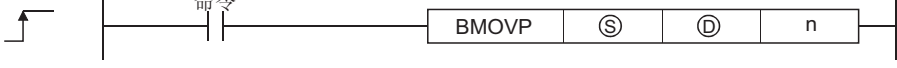
步	指令	软元件
0	LD	X3
1	DCMLP	D0
		D16
4	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.4.5 块 16 位数据传送(BMOV 与 BMOVP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
Ⓔ	○						—	—	
Ⓕ	○						—	—	
n	○						○	—	

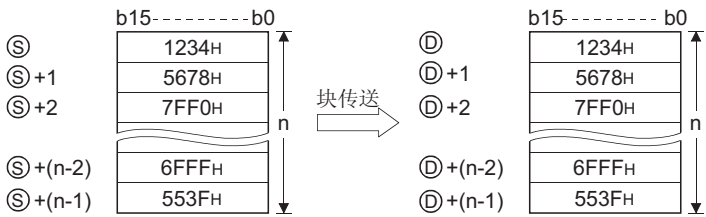
指令符号	执行条件
BMOV	
BMOVP	

[设定数据]

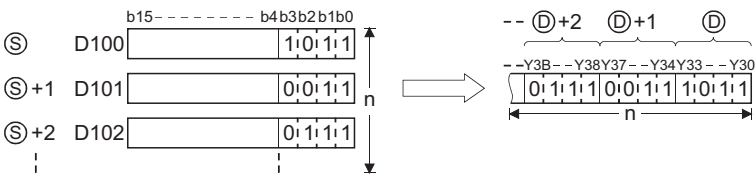
设定数据	含义	数据类型
Ⓢ	存储将要传送的数据的起始软元件号	BIN 16 位
Ⓓ	目标起始软元件号	
n	传输的数量(如果使用特殊的直接软元件(U\G), 则传输数量为 1~6144(QnACPU))	

[功能]

- (1) 将从Ⓢ指定的软元件开始的软元件中的 n 点 16 位数据批量传送到从Ⓓ指定的软元件开始的 n 点位置。



- (2) 即使源软元件和目标软元件之间存在相互重叠的情况，传送也照样能够完成。
如果传送到较小的软元件号，则从Ⓢ开始传送；如果传送到较大的软元件号，则从Ⓢ+(n-1)开始传送。
- (3) 当Ⓢ是字软元件而Ⓓ是位软元件时，字软元件的对象将是由位软元件的数字标识所指定的位数。
如果Ⓓ指定了 K1Y30，那么Ⓢ指定的字软元件的低四位将成为该对象。



- (4) 如果给⑤和⑥指定的都是位软元件，那么⑤和⑥应该总是具有相同的数字位数。
- (5) 在⑤或⑥中，只有一个可以指定给 MELSECNET/10 (H) 直接软元件和智能化功能模块/特殊功能模块。

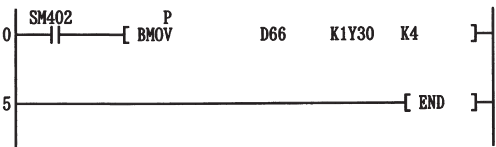
[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 从⑤或⑥起始的 n 点软元件的范围超出了相关软元件的范围。 (错误代码：4101)
 - 使用特殊直接软元件时，传输数量超过了 6144。 (QnACPU) (错误代码：4101)

[程序示例]

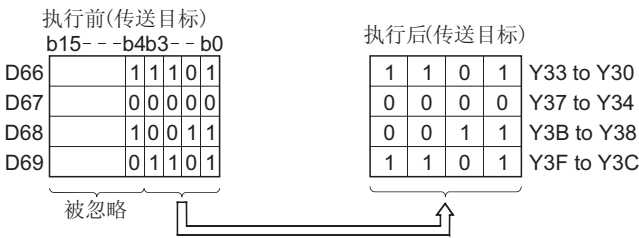
(1) 以下程序将 D66 到 D69 中数据的低 4 位输出到 4 点单元 Y30 到 Y3F。

[梯形图模式]



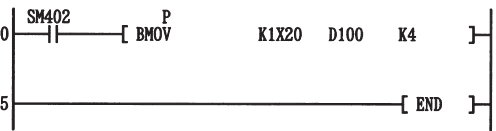
[列表模式]

步	指令	软元件
0	LD	SM402
1	BMOVP	D66 K1Y30 K4
5	END	



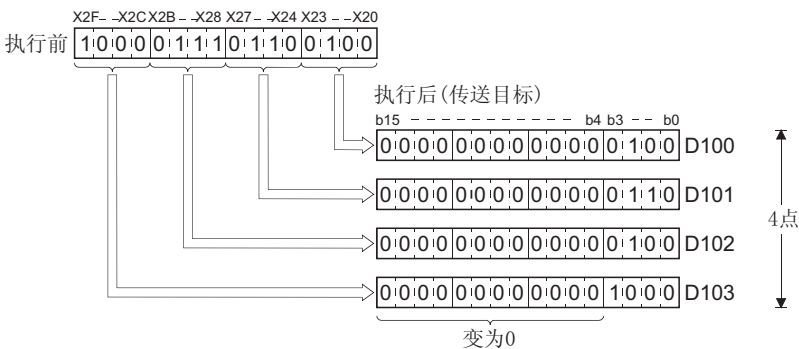
(2) 以下程序将 X20 到 X2F 中的数据输出到 4 点单元 D100 到 D103。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM402
1	BMOVP	K1X20 D100 K4
5	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.4.6 相同的 16 位数据块传送(FMOV 与 FMOVP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
Ⓔ	○						○		—
Ⓕ	○						—		—
n	○						○		—

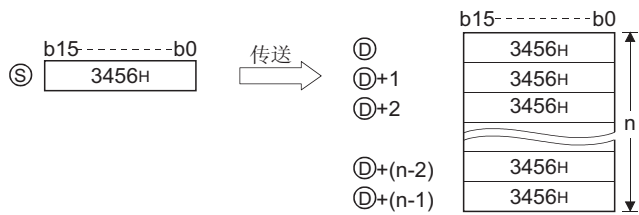
[指令符号]	[执行条件]
FMOV	
FMOVP	

[设定数据]

设定数据	含义	数据类型
Ⓢ	要传送的数据或存储要传送的数据的起始软元件号	BIN 16 位
Ⓓ	目标起始软元件号	
n	传输的数量（如果使用特殊的直接软元件(U\G)，则传输数量为 1~6144(QnACPU)）	

[功能]

(1) 将Ⓢ指定的软元件中的 16 位数据传送到从Ⓓ指定的软元件开始的 n 点位置。



(2) 如果Ⓢ指定一个字软元件而Ⓓ指定一个位软元件，那么由位软元件的数字标识指定的位数就是字软元件的对象位。

如果Ⓓ指定了 K1Y30，那么Ⓢ指定的字软元件的对象位将是低 4 位。



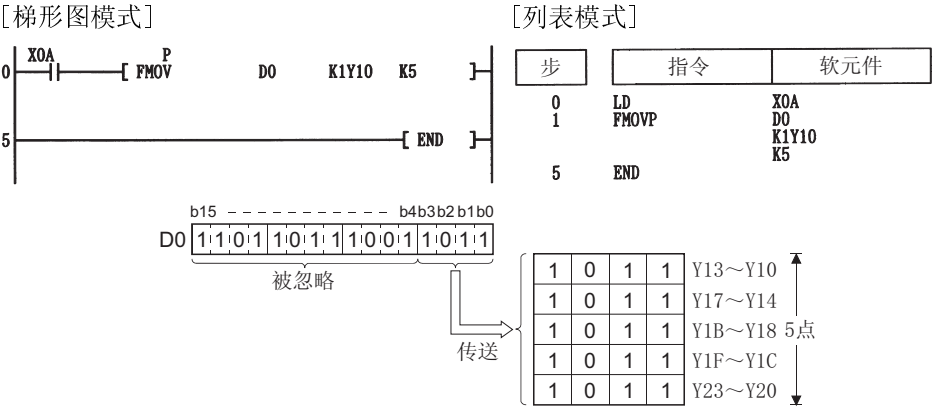
(3) 如果给Ⓢ和Ⓓ指定的都是位软元件，那么Ⓢ和Ⓓ应该总是具有相同的数字位数。

[运行错误]

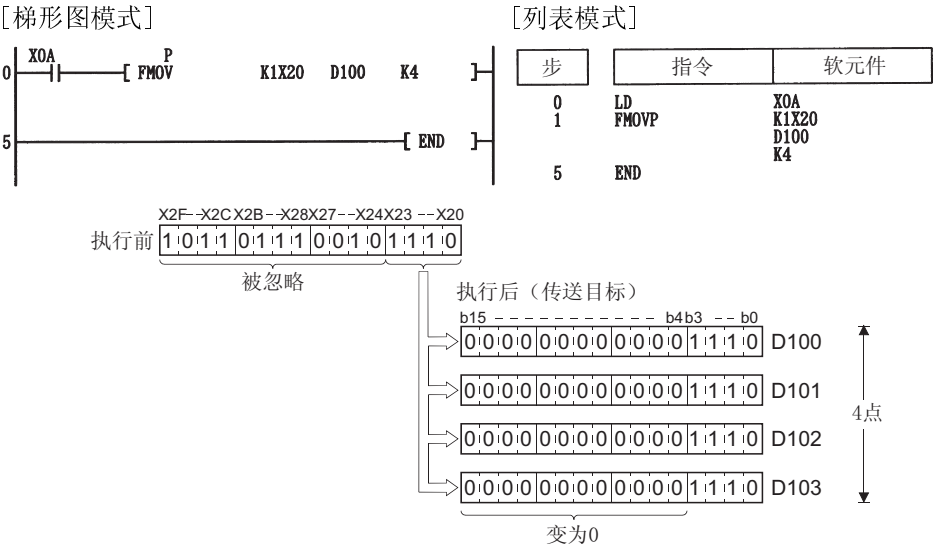
- (1) 在发生以下运行错误的情况中，错误标志（SM0）变为 ON，并且错误代码存储在 SD0 中。
- 从Ⓓ起始的 n 点软元件的范围超出了该软元件范围。 (错误代码：4101)
 - 使用特殊直接软元件时，传输数量超过了 6144。 (QnACPU) (错误代码：4101)

[程序示例]

(1) 当 XA 变为 ON 时，以下程序将 D0 中数据的低 4 位输出到 Y10 到 Y23。



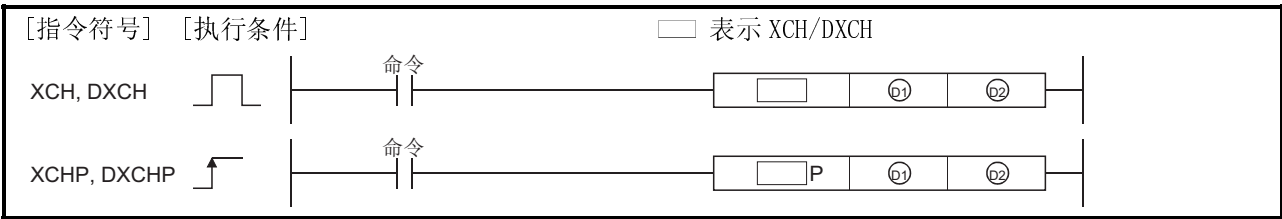
(2) 当 XA 变为 ON 时，以下程序将 X20 到 X23 中的数据输出到 D100 到 D103。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.4.7 16 位和 32 位数据交换 (XCH, XCHP, DXCH 与 DXCHP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
①	○						—		
②	○						—		



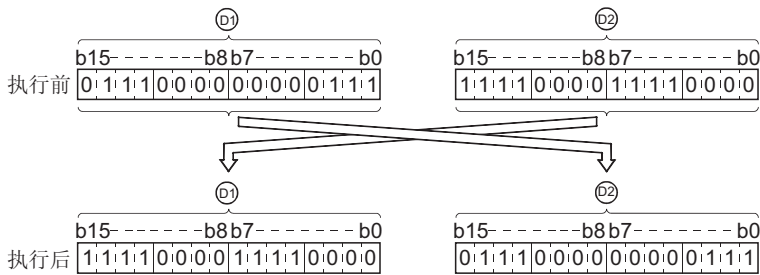
[设定数据]

设定数据	含义	数据类型
①	存储将进行交换的数据的起始软元件号	BIN 16/32 位
②		

[功能]

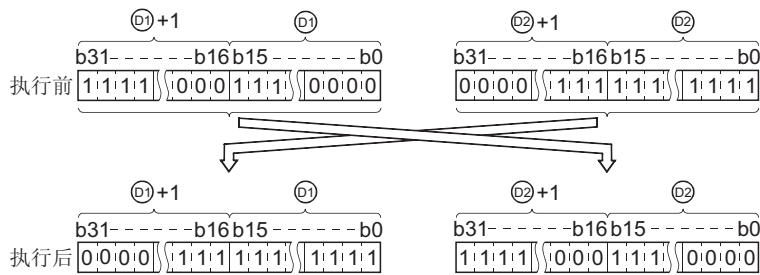
XCH

(1) 在①和②之间执行 16 位数据交换。



DXCH

(1) 在①+1 和②+2 之间执行 32 位数据交换。



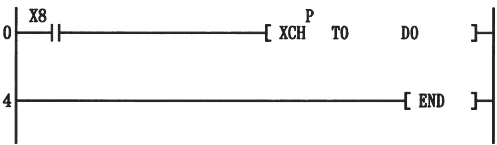
[运行错误]

(1) 不存在与 XCH(P) 和 DXCH(P) 指令相关的运行错误。

[程序示例]

(1) 当 X8 变为 ON 时，以下程序将 T0 的当前值与 D0 的内容进行交换。

[梯形图模式]

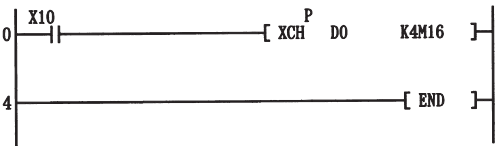


[列表模式]

步	指令	软元件
0	LD	X8
1	XCHP	T0 D0
4	END	

(2) 当 X10 变为 ON 时，以下程序将 D0 的内容与 M16 到 M31 中的数据进行交换。

[梯形图模式]

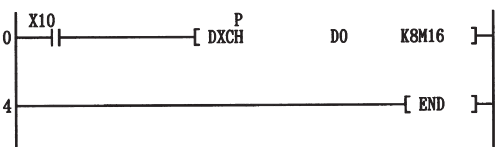


[列表模式]

步	指令	软元件
0	LD	X10
1	XCHP	D0 K4M16
4	END	

(3) 当 X10 变为 ON 时，以下程序将 D0 和 D1 的内容与 M16 到 M47 中的数据进行交换。

[梯形图模式]

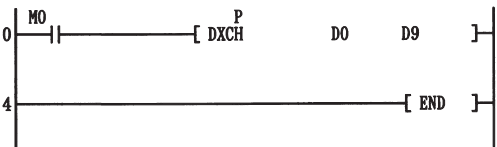


[列表模式]

步	指令	软元件
0	LD	X10
1	DXCHP	D0 D1 K8M16
4	END	

(4) 当 M0 变为 ON 时，以下程序将 D0 和 D1 的内容与 D9 和 D10 的内容进行交换。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	M0
1	DXCHP	D0 D1 D9 D10
4	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.4.8 块 16 位数据交换 (BXCH 与 BXCHP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
①	—	○			—				—
②	—	○			—				—
n	○	○			○				—

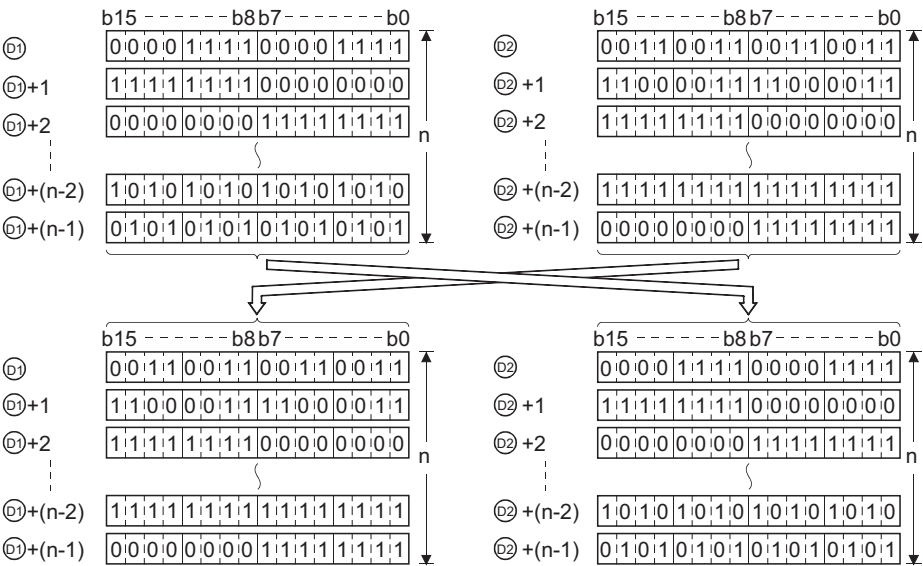
[指令符号]	[执行条件]
BXCH	
BXCHP	

[设定数据]

设定数据	含义	数据类型
①	存储将进行交换的数据的软元件的起始号	BIN 16 位
②		
n	交换数	

[功能]

(1) 将从①指定的软元件开始的软元件中的 n 点 16 位数据与从②指定的软元件开始的软元件中的 n 点 16 位数据进行交换。



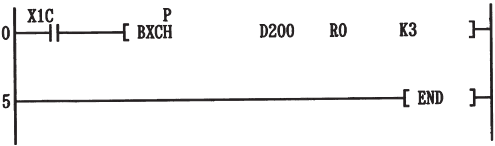
[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 从⑥①或⑥②软元件开始的 n 点软元件范围超出了相关软元件。 (错误代码：4101)
 - ⑥①和⑥②软元件相互重叠。 (错误代码：4101)

[程序示例]

- (1) 当 X1C 变为 ON 时，以下程序将从 D200 起始的 3 点 16 位数据与从 R0 起始的 3 点 16 位数据进行交换。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X1C
1	BXCHP	D200 R0 K3
5	END	

	b15	-----	b8	b7	-----	b0
D200	0	0	1	1	1	1
D201	1	1	1	1	1	1
D202	1	0	1	0	1	0

	b15	-----	b8	b7	-----	b0
R0	0	1	1	1	0	1
R1	0	0	1	1	0	1
R2	1	1	0	0	0	0

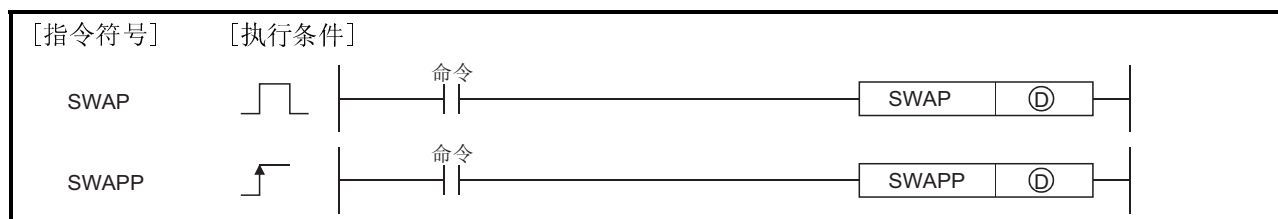
	b15	-----	b8	b7	-----	b0
D200	0	1	1	1	0	1
D201	0	0	1	1	0	1
D202	1	1	0	0	0	0

	b15	-----	b8	b7	-----	b0
R0	0	0	1	1	1	0
R1	1	1	1	1	1	0
R2	1	0	1	0	1	0

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

6.4.9 高字节和低字节交换 (SWAP 与 SWAPP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J□N□		特殊功能模 块 U□G□	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
④	○						—		

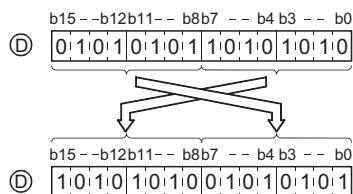


「設定データ」

设定数据	含义	数据类型
⑤	存储数据的起始软元件号	BIN 16 位

〔功能〕

- (1) 将⑤指定的软元件的高 8 位和低 8 位进行交换。

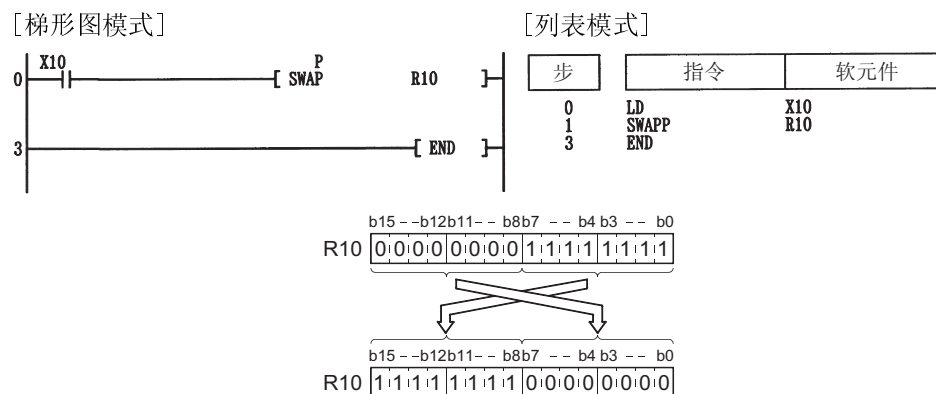


[运行错误]

- (1) 不存在与 SWAP(P) 指令相关的运行错误。

[程序示例]

- (1) 当 X10 变为 ON 时, 以下程序将 R10 的高 8 位和低 8 位进行交换。

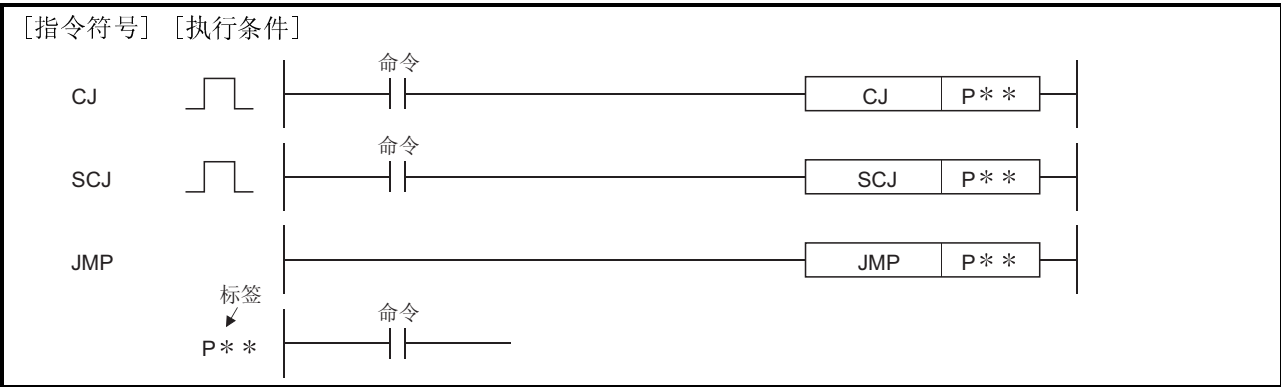


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.5 程序分支指令

6.5.1 指针分支指令 (CJ, SCJ, JMP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][] G[][]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				P
P	—								○



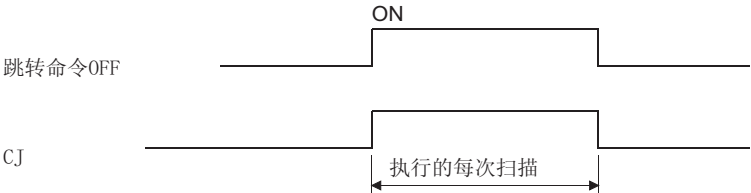
[设定数据]

设定数据	含义	数据类型
P**	跳转目的地的指针号码	软元件名称

[功能]

CJ

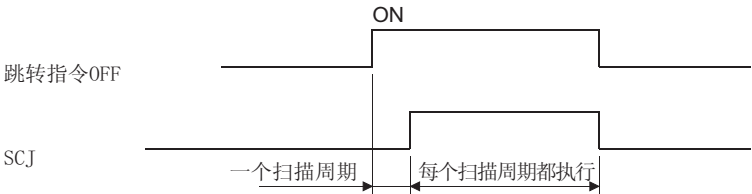
- (1) 当跳转命令为 ON 时，在同一程序文件内执行指定指针号码的程序。
- (2) 当跳转命令为 OFF 时，执行程序的下一步。



SCJ

- (1) 当跳转命令从 OFF 变为 ON 时，从下一次扫描开始在同一程序文件内执行指定指针号码的程序。

(2) 当跳转命令为 OFF 或者从 ON 变为 OFF 时，执行程序的下一步。



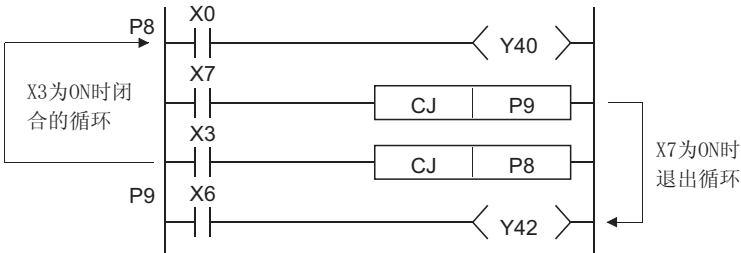
JMP

(1) 在同一程序文件内无条件地执行指针号码所指定的程序。

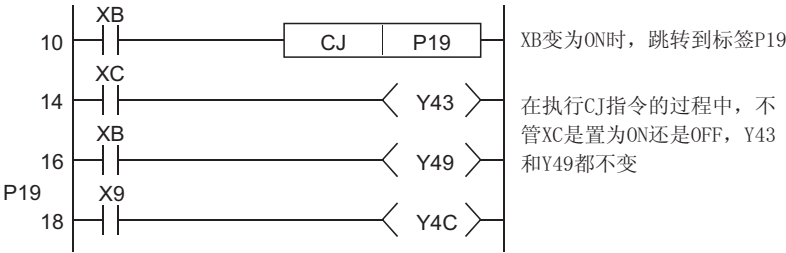
要点

- 使用跳转指令时，请注意以下要点。
- (1) 在定时器线圈已经变为 ON 后，如果试图使用 CJ、SCJ 或 JMP 指令来跳转已经变为 ON 的线圈定时器，那么进行精确测量是不可能的。
 - (2) 如果使用 CJ、SCJ 或 JMP 指令强制跳转到 OUT 指令，那么扫描时间会缩短。
 - (3) 如果使用 CJ、SCJ 或 JMP 指令强制跳转到程序尾部，那么扫描时间会缩短。
 - (4) CJ、SCJ 或 JMP 指令可用于跳转到当前正执行的步骤之前的某一步。

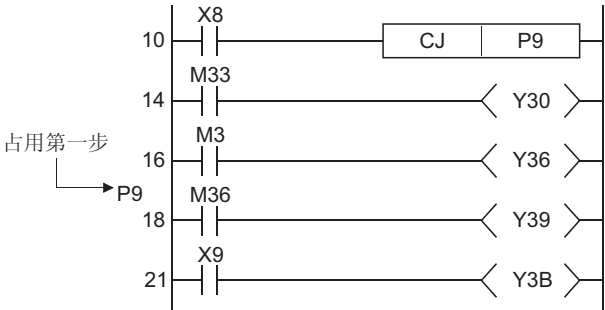
然而，有必要考虑退出循环的方法，以便在该过程中看门狗定时器不会超时。



(5) 已通过 CJ、SCJ 或 JMP 实现了的跳转的软元件不发生变化。



(6) 标签(P*) 占用第一步。



- (7) 跳转指令只可用于同一程序文件内的指针号码。
- (8) 在跳转运算期间，如果跳转到跳转范围内的一个指针号码，那么程序将从跳转目标的指针号码开始继续执行。

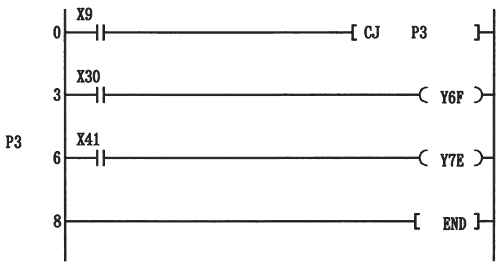
[运行错误]

- (1) 在以下情况中会返回运行错误，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 在 END 指令之前没有出现指定的指针号码。(错误代码：4210)
 - 在同一程序里已经指定了一个并不用作标签的指针号码。(错误代码：4210)
 - 已经指定了一个公共指针。(错误代码：4210)

[程序示例]

(1) 当 X9 变为 ON 时，以下程序跳转到 P3。

[梯形图模式]

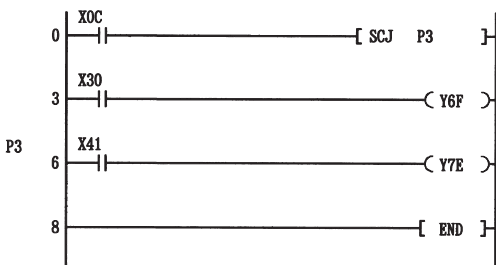


[列表模式]

步	指令	软元件
0	LD	X9
1	CJ	P3
3	LD	X30
4	OUT	Y6F
5		P3
6	LD	X41
7	OUT	Y7E
8	END	

(2) 在 XC 变为 ON 后，以下程序从下一次扫描开始跳转到 P3。

[梯形图模式]



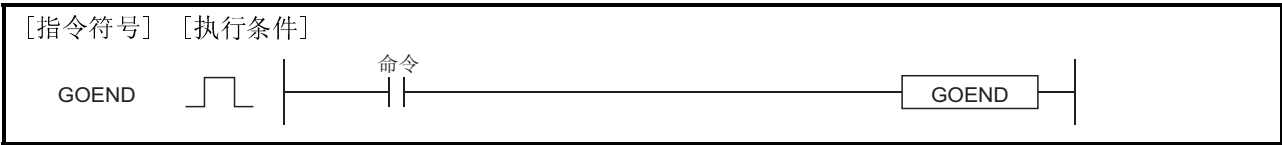
[列表模式]

步	指令	软元件
0	LD	X0C
1	SCJ	P3
3	LD	X30
4	OUT	Y6F
5		P3
6	LD	X41
7	OUT	Y7E
8	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.5.2 跳转到 END (GOEND)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][6][]	变址寄存器 Zn	常量 H, K	其它
	位	字		位	字				
—	—								



[功能]

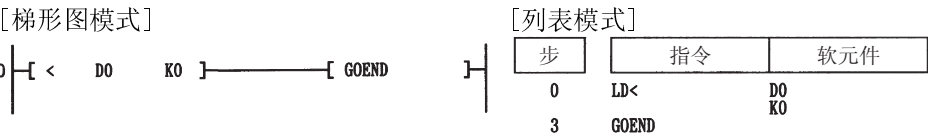
(1) 跳转到同一程序文件内的 FEND 或 END 指令。

[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SMO) 为 ON，并且错误代码存储在 SD0 中。
- 在执行 CALL 和 ECALL 指令之后并在执行 RET 指令之前已经执行了 GOEND 指令。(错误代码: 4211)
 - 在执行 FOR 指令之后并在执行 NEXT 指令之前已经执行了 GOEND 指令。(错误代码: 4200)
 - 在执行中断程序期间，已经在执行 IRET 指令之前执行了 GOEND 指令。(错误代码: 4221)
 - 在 CHKCIR 和 CHKEND 指令块之间执行了 GOEND 指令。(错误代码: 4230)
 - 在 IX 和 IXEND 指令块之间执行了 GOEND 指令。(错误代码: 4231)

[序示例]

(1) 如果 D0 容纳的是一个负数，那么以下程序跳转到 END 指令。



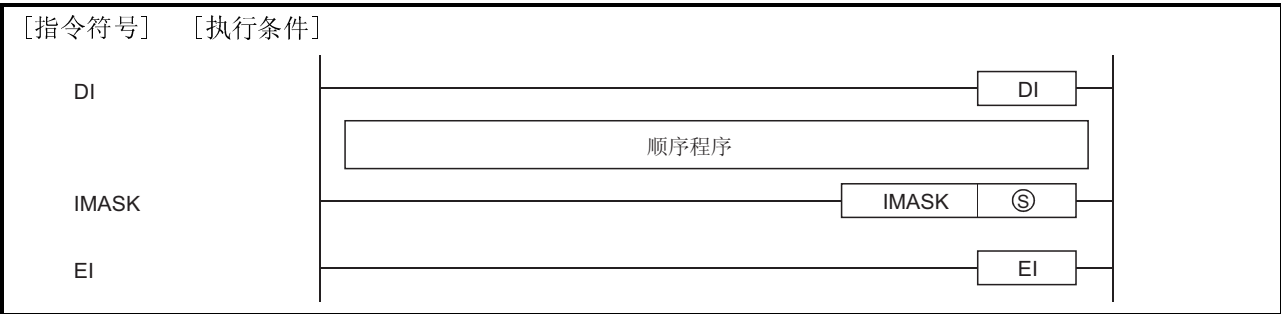
QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.6 程序执行控制指令

6.6.1 中断禁止/允许指令，中断程序屏蔽(DI, EI, IMASK)

(1) 使用基本型 QCPU 时

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	—	○	—						



[设定数据]

设定数据	含义	数据类型
⑤	中断屏蔽数据，或者存储中断屏蔽数据的起始号的软元件	BIN 16 位

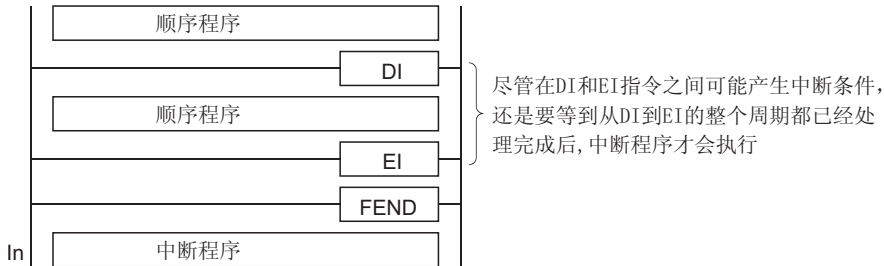
[功能]

DI

- (1) 即使出现了中断程序的启动条件，也只能在执行了 EI 指令之后才允许执行中断程序。
- (2) 当电源为 ON 或者复位 CPU 模块时，输入一个 DI 状态。

EI

EI 指令用于清除由于执行 DI 指令而引起的中断禁止状态，并创建一个状态；在该状态下，可以执行由 IMASK 指令确认的中断指针号码所指定的中断程序。
如果 IMASK 指令不执行，则禁止 I32 到 I47。



IMASK

- (1) 允许或禁止执行使用从⑤指定的软元件开始的 8 点的位模式指定的中断指针来标记的中断程序。
- 1 (ON)..... 允许执行中断程序
 - 0 (OFF)..... 禁止执行中断程序

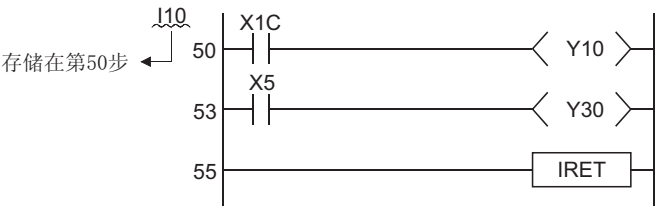
(2) 对应于单个位的中断指针号码如下所示：

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
⑤	I15	I14	I13	I12	I11	I10	I9	I8	I7	I6	I5	I4	I3	I2	I1	I0
⑤ + 1	I31	I30	I29	I28	I27	I26	I25	I24	I23	I22	I21	I20	I19	I18	I17	I16
⑤ + 2	I47	I46	I45	I44	I43	I42	I41	I40	I39	I38	I37	I36	I35	I34	I33	I32
⑤ + 3	I63	I62	I61	I60	I59	I58	I57	I56	I55	I54	I53	I52	I51	I50	I49	I48
⑤ + 4	I79	I78	I77	I76	I75	I74	I73	I72	I71	I70	I69	I68	I67	I66	I65	I64
⑤ + 5	I95	I94	I93	I92	I91	I90	I89	I88	I87	I86	I85	I84	I83	I82	I81	I80
⑤ + 6	I111	I110	I109	I108	I107	I106	I105	I104	I103	I102	I101	I100	I99	I98	I97	I96
⑤ + 7	I127	I126	I125	I124	I123	I122	I121	I120	I119	I118	I117	I116	I115	I114	I113	I112

- (3) 当电源为 ON 或者通过执行中断程序 I0 到 I31 复位 CPU 模块时，I48 到 I127 有效。
- (4) 软元件⑤、⑤+1、⑤+2 以及⑤+3 到⑤+7 的状态存储在 SD715 到 SD717 和 SD781 到 SD785 (IMASK 指令屏蔽模式的存储区域) 中。
- (5) 尽管特殊寄存器被分隔为 SD715 到 SD717 和 SD781 到 SD785，软元件号应该被连续地指定为从⑤到⑤+7。

要点

(1) 一个中断指针占用一步。



- (2) 关于中断条件的信息，请参照基本型 QCPU (Q 模式) 用户手册 (功能解释编程基础)。
- (3) 在中断程序执行期间，DI 状态 (中断禁止) 是激活的。
不要在中断程序中插入 EI 指令以试图执行多个中断，即在中断程序内部嵌套运行中断程序。
- (4) 如果 EI 和 DI 指令位于主控制中，那么不管 MC 指令处于执行还是非执行状态，这些指令都将执行。

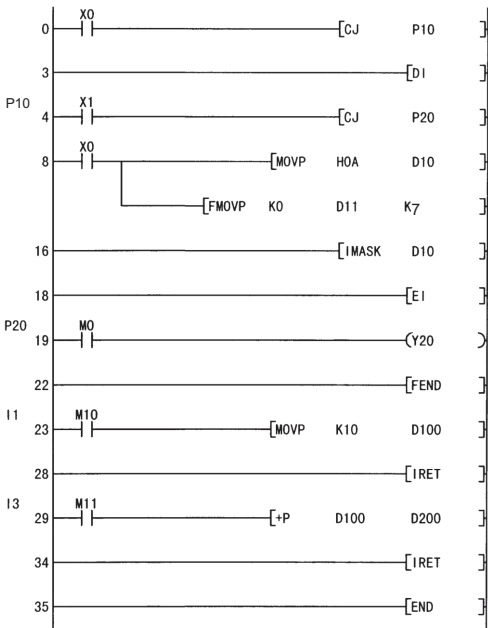
[运行错误]

- (1) 不存在与 DI 和 EI 指令相关的运行错误。
- (2) 不存在与 IMASK 指令相关的运行错误。

[程序示例]

(1) 以下程序旨在当 X0 为 ON 时，只允许中断指针号码为 I1 和 I3 的中断程序执行。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	CJ	P10
3	DI	
4	P10	
5	LD	X1
6	CJ	P20
8	LD	X0
9	MOV P	HOA D10
12	FMOV P	K0 D11 K7
16	IMASK	D10
18	EI	
19	P20	
20	LD	MO
21	OUT	Y20
22	FEND	
23	I1	
24	LD	M10
25	MOV P	K10 D100
28	IRET	
29	I3	
30	LD	M11
31	+P	D100 D200
34	IRET	
35	END	

(2) 使用高性能型 QCPU/过程控制 CPU 时

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
Ⓢ	—	○					—



[设定数据]

设定数据	含义	数据类型
Ⓢ	中断屏蔽数据或者存储中断屏蔽数据的起始号软元件	BIN 16 位

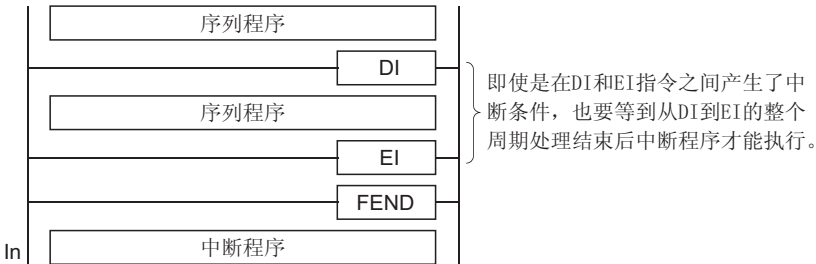
[功能]

DI

- (1) 即使出现了中断程序启动条件，也只能在执行了 EI 指令之后才允许执行中断程序。
- (2) DI 状态在电源变为 ON 或者复位 CPU 模块时输入。

EI

EI 指令用于清除由于执行 DI 指令而引起的中断禁止状态，并创建一个状态；在该状态下，可以执行由 IMASK 指令确认的中断指针号码所指定的中断程序，也可以执行固定周期执行类型程序。如果 IMASK 指令不执行，则禁止 I32 到 I47。



IMASK

- (1) 允许或禁止执行使用从⑤指定的软元件开始的 16 点位模式所指定的中断地址标记的中断程序。
- 1 (ON) 允许执行中断程序
 - 0 (OFF) 禁止执行中断程序

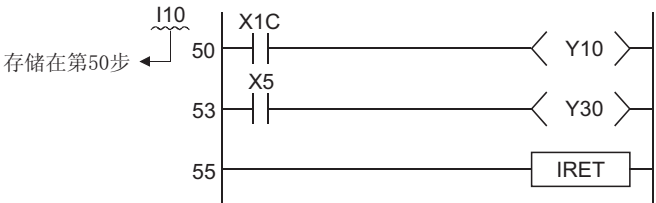
(2) 对应于单个位的中断指针号码如下所示：

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
⑤	I15	I14	I13	I12	I11	I10	I9	I8	I7	I6	I5	I4	I3	I2	I1	I0
⑤ + 1	I31	I30	I29	I28	I27	I26	I25	I24	I23	I22	I21	I20	I19	I18	I17	I16
⑤ + 2	I47	I46	I45	I44	I43	I42	I41	I40	I39	I38	I37	I36	I35	I34	I33	I32
⑤ + 3	I63	I62	I61	I60	I59	I58	I57	I56	I55	I54	I53	I52	I51	I50	I49	I48
⑤ + 4	I79	I78	I77	I76	I75	I74	I73	I72	I71	I70	I69	I68	I67	I66	I65	I64
⑤ + 5	I95	I94	I93	I92	I91	I90	I89	I88	I87	I86	I85	I84	I83	I82	I81	I80
⑤ + 6	I111	I110	I109	I108	I107	I106	I105	I104	I103	I102	I101	I100	I99	I98	I97	I96
⑤ + 7	I127	I126	I125	I124	I123	I122	I121	I120	I119	I118	I117	I116	I115	I114	I113	I112
⑤ + 8	I143	I142	I141	I140	I139	I138	I137	I136	I135	I134	I133	I132	I131	I130	I129	I128
⑤ + 9	I159	I158	I157	I156	I155	I154	I153	I152	I151	I150	I149	I148	I147	I146	I145	I144
⑤ + 10	I175	I174	I173	I172	I171	I170	I169	I168	I167	I166	I165	I164	I163	I162	I161	I160
⑤ + 11	I191	I190	I189	I188	I187	I186	I185	I184	I183	I182	I181	I180	I179	I178	I177	I176
⑤ + 12	I207	I206	I205	I204	I203	I202	I201	I200	I199	I198	I197	I196	I195	I194	I193	I192
⑤ + 13	I223	I222	I221	I220	I219	I218	I217	I216	I215	I214	I213	I212	I211	I210	I209	I208
⑤ + 14	I239	I238	I237	I236	I235	I234	I233	I232	I231	I230	I229	I228	I227	I226	I225	I224
⑤ + 15	I255	I254	I253	I252	I251	I250	I249	I248	I247	I246	I245	I244	I243	I242	I241	I240

- (3) 当电源为 ON 或者复位 CPU 模块时，允许执行中断程序 I0 到 I31 和 I48 到 I127，但禁止执行中断程序 I32 到 I47。
- (4) 软元件⑤、⑤+1、⑤+2 以及⑤+3 到⑤+15 的状态存储在 SD715 到 SD717 和 SD781 到 SD793 (IMASK 指令屏蔽模式的存储区域) 中。
- (5) 尽管特殊寄存器分隔为 SD715 到 SD717 和 SD781 到 SD793，软元件号应该被连续地指定为从⑤到⑤+15。

要点

- (1) 一个中断指针占用一步。



- (2) 有关中断条件的信息，请参照使用中的 CPU 模块用户手册 (功能说明和编程基础)。
- (3) 在中断程序执行期间，DI 状态 (中断禁止) 是激活的。不要在中断程序中插入 EI 指令以试图执行多个中断，即在中断程序内部嵌套运行中断程序。
- (4) 如果 EI 和 DI 指令位于主控制中，那么不管 MC 指令处于执行还是非执行状态，这些指令都将执行。

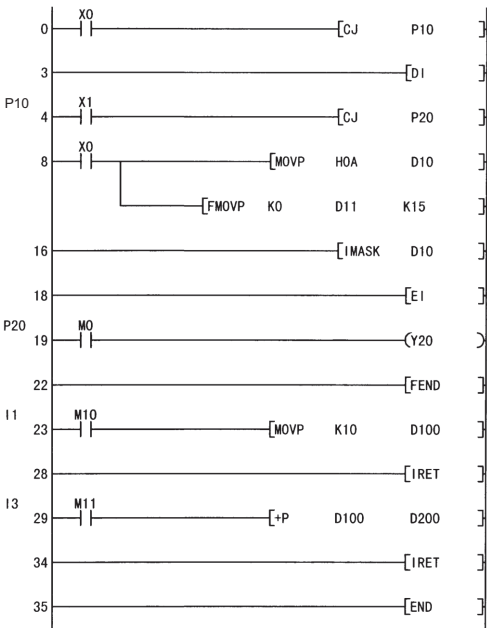
[运行错误]

- (1) 不存在与 DI 和 EI 指令相关的运行错误。
- (2) 不存在与 IMASK 指令相关的运行错误。

[程序示例]

(1) 当 X0 为 ON 时，以下程序为中断指针号码标记的中断程序创建允许执行状态。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	CJ	P10
3	DI	
4	P10	
5	LD	X1
6	CJ	P20
8	LD	X0
9	MOV P	HOA D10
12	FMOV P	K0 D11 K15
16	IMASK	D10
18	EI	
19	P20	
20	LD	M0
21	OUT	Y20
22	FEND	
23	I1	
24	LD	M10
25	MOV P	K10 D100
28	IRET	
29	I3	
30	LD	M11
31	+P	D100 D200
34	IRET	
35	END	

(3) 使用 QnACPU 时

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		变址寄存器 U[][]	变址寄存器 Zn	常量 H, K	其它
	位	字		位	字				
⑤	—	○	—						



[设定数据]

设定数据	含义	数据类型
Ⓢ	中断屏蔽数据或者存储中断屏蔽数据的起始号软元件	BIN 16 位

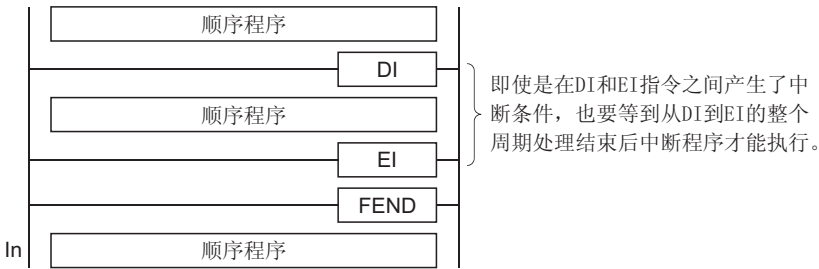
[功能]

DI

- (1) 即使出现了中断程序启动条件，也只能在执行了 EI 指令之后才允许执行中断程序。
- (2) 当电源变为 ON 或者复位 CPU 模块时，输入一个 DI 状态。

EI

EI 指令用于清除由于执行 DI 指令而引起的中断禁止状态，并创建一个状态；在该状态下，可以执行由 IMASK 指令确认的中断指针号码所指定的中断程序。
如果 IMASK 指令不执行，则禁止 I32 到 I47。



IMASK

- (1) 允许或禁止执行使用从⑤指定的软元件开始的 3 点位模式所指定的中断地址标记的中断程序。
- 1 (ON) 允许执行中断程序
 - 0 (OFF) 禁止执行中断程序

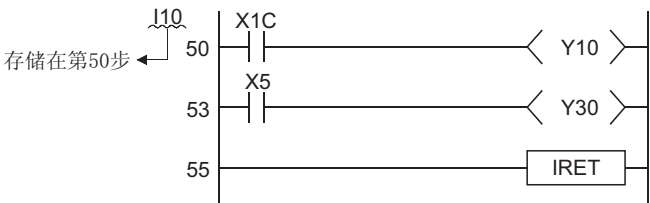
(2) 对应于单个位的中断指针号码如下所示：

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
⑤	I15	I14	I13	I12	I11	I10	I9	I8	I7	I6	I5	I4	I3	I2	I1	I0
⑤ + 1	I31	I30	I29	I28	I27	I26	I25	I24	I23	I22	I21	I20	I19	I18	I17	I16
⑤ + 2	I47	I46	I45	I44	I43	I42	I41	I40	I39	I38	I37	I36	I35	I34	I33	I32

- (3) 当电源变为 ON 或者复位 CPU 模块时，中断程序 I0 到 I31 处于允许执行状态，而中断程序 I32 到 I147 处于禁止执行状态。
- (4) 软元件⑤、⑤+1、⑤+2 的状态存储在 SD715 到 SD717 (IMASK 指令屏蔽模式的存储区域) 中。

要点

(1) 一个中断指针占用一步。



- (2) 有关中断条件的信息，请参照 QnACPU 编程手册 (基本原理)。
- (3) 在中断程序执行期间，DI 状态 (中断禁止) 是激活的。不要在中断程序中插入 EI 指令以试图执行多个中断，即在中断程序内部嵌套运行中断程序。
- (4) 如果 EI 和 DI 指令位于主控制中，那么不管 MC 指令处于执行还是非执行状态，这些指令都将执行。

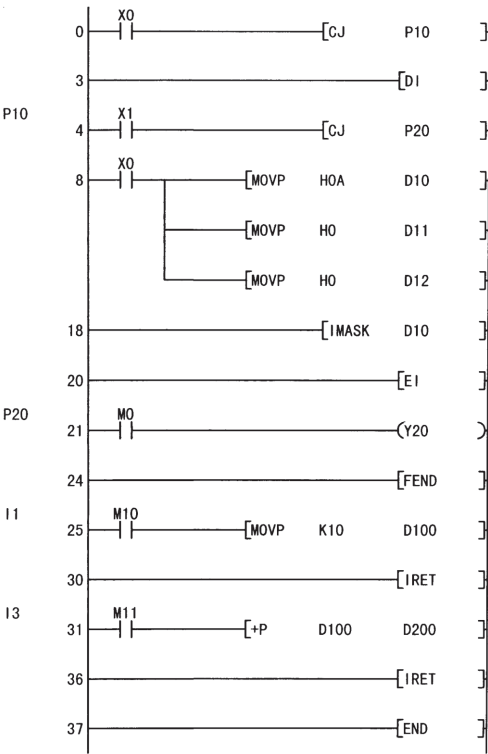
[运行错误]

- (1) 不存在与 DI 和 EI 指令相关的运行错误。
- (2) 不存在与 IMASK 指令相关的运行错误。

[程序示例]

(1) 当 X0 为 ON 时，以下程序为中断指针号码标记的中断程序创建允许执行状态。

[梯形图模式]



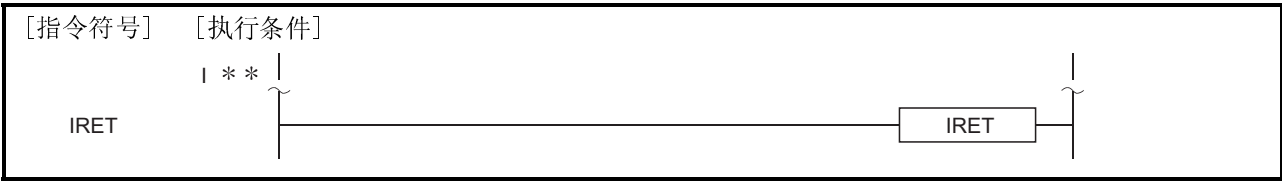
[列表模式]

步	指令	软元件
0	LD	X0
1	CJ	P10
3	DI	
4	P10	
5	LD	X1
6	CJ	P20
8	LD	X0
9	MOV	HOA D10
12	MOV	HO D11
15	MOV	HO D12
18	IMASK	D10
20	EI	
21	P20	
22	LD	M0
23	OUT	Y20
24	FEND	
25	I1	
26	LD	M10
27	MOV	K10 D100
30	IRET	
31	I3	
32	LD	M11
33	+P	D100 D200
36	IRET	
37	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.6.2 从中断程序中恢复 (IRET)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J K L		特殊功能模 块 U V G	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
—	—								



[功能]

- (1) 表示对中断程序的处理完成。
- (2) 执行 IRET 指令后，返回到对顺序程序的处理。

[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 为 ON，并且错误代码存储在 SD0 中。
- 没有与中断地址相对应的指针。

(错误代码: 4220)

● 在执行中断程序之前已经先发出了 IRET 指令。

(错误代码: 4223)

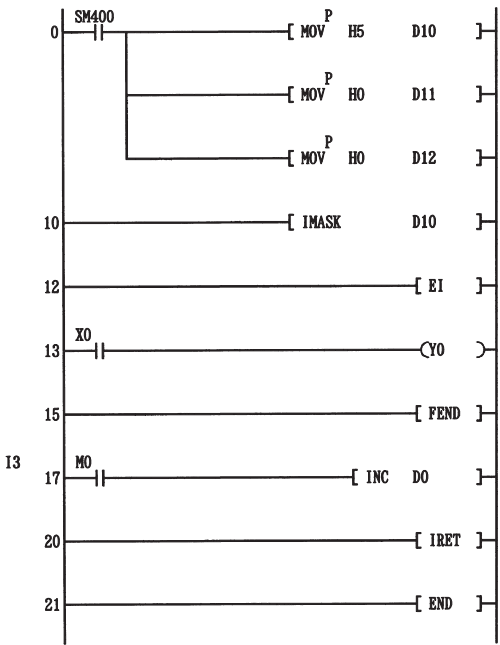
● 在产生中断之后而在执行 IRET 指令之前已经先执行了 END、FEND、GOEND 或 STOP 指令。

(错误代码: 4221)

[程序示例]

(1) 在产生 3 号中断的时候，如果 M0 为 ON，则以下程序给 D0 加 1。

[梯形图模式]



[列表模式]

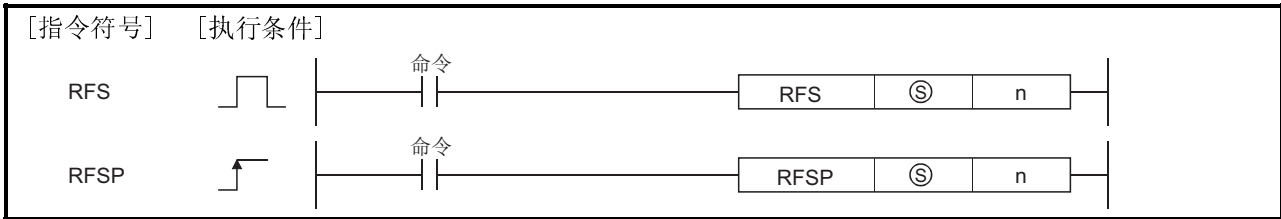
步	指令	软元件
0	LD	SM400
1	MOVP	H5 D10
4	MOVP	H0 D11
7	MOVP	H0 D12
10	IMASK	D10
12	EI	
13	LD	X0
14	OUT	Y0
15	FEND	
16		I3
17	LD	M0
18	INC	D0
20	IRET	
21	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

6.7 I/O 刷新指令

6.7.1 I/O 刷新(RFS 与 RFSP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\C[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
Ⓢ	○ (仅限于 X, Y)	—							—
n	○	○							—



[设定数据]

设定数据	含义	数据类型
Ⓢ	执行刷新操作的软元件的起始软元件号	位
n	要刷新的点数	BIN 16 位

[功能]

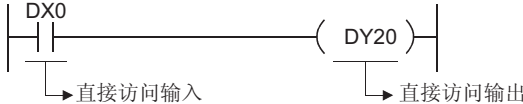
- (1) 只对在一个扫描周期期间扫描过的软元件进行刷新，并从外部源获取输入数据或者将数据输出到输出模块。
- (2) 从外部源获取输入数据或者将数据输出到外部源只在执行了 END 指令之后批量执行，所以，不可能在一个扫描周期的执行期间将脉冲信号输出到外部源。
执行 I/O 刷新指令的时候，程序在执行过程中对相应软元件号的输入(X)或输出(Y)进行强制刷新。因此，可以在一个扫描周期期间将脉冲信号输出到外部源。

(3) 使用直接访问输入 (DX) 或直接访问输出 (DY) 以 1 点为单元刷新输入 (X) 或输出 (Y)。

[基于RFS指令的程序]



[基于直接访问输入和直接访问输出的程序]



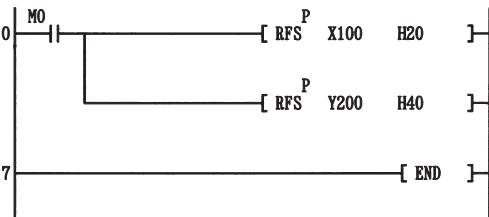
[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 置为 ON，并且错误代码存储在 SD0 中。
- 从⑤指定的软元件开始的 n 点范围超出了最近的范围。 (错误代码：4101)

[程序示例]

(1) 当 M0 变为 ON 时，以下程序对 X100 到 X11F 和 Y200 到 Y23F 进行刷新。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	M0
1	RFSP	X100 H20
4	RFSP	Y200 H40
7	END	

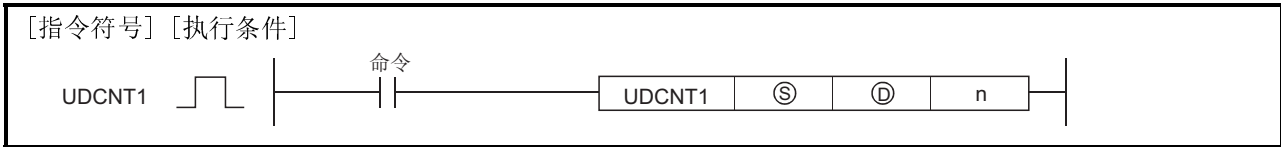
QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

6.8 其它方便的指令

6.8.1 单相输入加法或减法计数器(UDCNT1)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\G[]		特殊功能模 块 []G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
Ⓢ	○ (限于 X)	—	—	—					—
Ⓣ	—	△ * (限于 C)	—	—					—
n	△ *	△ *	△ *	○					—

*:不能使用局部软元件和为个别程序设置的文件寄存器



[设定数据]

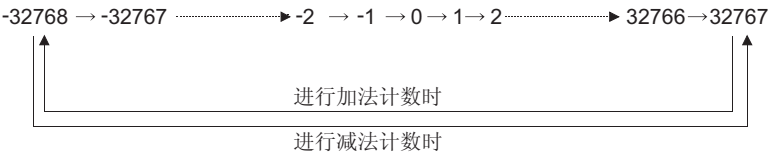
设定数据	含义	数据类型
Ⓢ	● Ⓢ+0: 用于计数输入的输入数字	位
	● Ⓢ+1: 用于设定加法、减法计数	
	● OFF : 加法计数 (计数时数字增加) ● ON : 减法计数 (计数时数字减少)	
Ⓓ	● 将根据 UDCNT1 指令进行计数的计数器号	软元件名称
n	● 设定值	BIN 16 位

[功能]

- (1) 当Ⓢ指定的输入从 OFF 变为 ON 时，Ⓓ指定的计数器的当前值将得到更新。
- (2) 计数的方向取决于Ⓢ+1 指定的输入状态是 ON 还是 OFF。
 - OFF : 加法计数 (以增加当前值的方式计数)
 - ON : 减法计数 (以减少当前值的方式计数)
- (3) 计数处理的执行方式描述如下：
 - 当进行加法计数时，如果当前值变得与 n 指定的设定值相同，则Ⓓ指定的计数器触点变为 ON。然而，即使是当Ⓓ指定的计数器触点变为 ON 时，对当前值的计数还将继续。(参见程序示例(1))
 - 当进行减法计数时，如果当前值变得比设定值少 1，则Ⓓ指定的计数器触点变为 OFF。(参见程序示例(1))
 - Ⓓ指定的计数器是环计数器。

在当前值为 32767 时，如果正进行加法计数，则当前值将变为-32768；另外，在当前值为-32768 时，如果正进行减法计数，则当前值将变为 32767。

基于当前值进行的计数处理如下所示：



- (4) 当计数命令从 OFF 变为 ON 时，基于 UDCNT1 指令的计数处理开始计数；当计数命令从 ON 变为 OFF 时，中止计数。当计数命令再一次从 OFF 变为 ON 时，从计数中止时的值重新开始计数。
- (5) RST 指令清除①指定的计数器的当前值并将触点置为 OFF。

要点

(1) UDCNT1 指令将参数软元件数据注册到 CPU 模块的工作区，而实际的计数运算被处理为系统中断。

(当命令输入置为 OFF 或者当 CPU 模块置为 STOP 然后又置为 RUN 时，将清除注册到 CPU 模块工作区的软元件数据)。

因此，可计数的脉冲的 ON 和 OFF 时间必须比 CPU 模块的中断间隔长。

CPU 模块的中断间隔如下所示。

CPU 模块类型名	中断间隔
高性能型 QCPU、过程控制 CPU	1ms
QnACPU	5ms

(2) 在基于 UDCNT1 指令的一次计数期间(当命令输入为 ON 时)，设定值不能更改。如果要更改设定值，请首先将命令输入置为 OFF。

(3) 其它指令不能使用 UDCNT1 指令指定的计数器。如果这些计数器为其它指令所使用，则它们不能返回精确的计数值。

(4) 在所有正执行的程序内部，UDCNT1 指令最多可以使用 6 次。

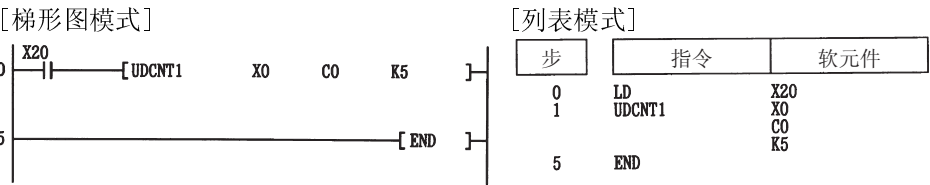
第七个以及以后的 UDCNT1 指令不会得到处理。

[运行错误]

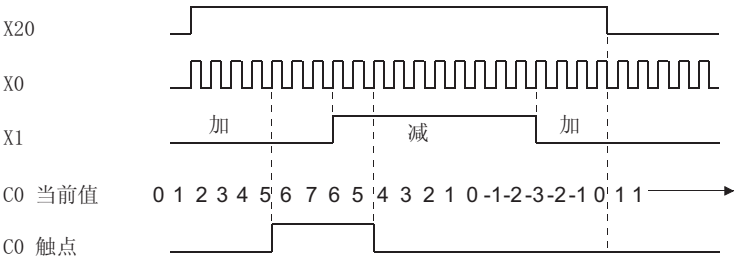
- (1) 不存在与 UDCNT1 指令相关的运行错误。

[程序示例]

- (1) 在 X20 变为 ON 后，该程序使用 C0(加法和减法计数器)对 X0 从 OFF 变为 ON 的次数进行计数。



[运算]

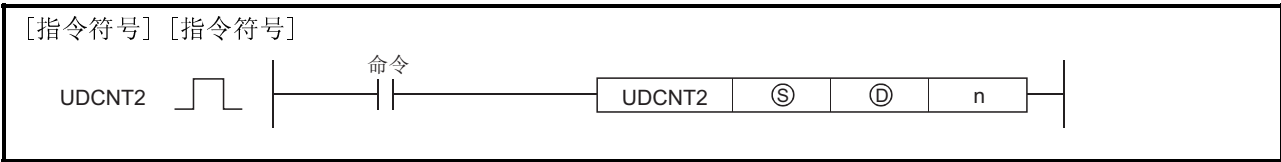


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

6.8.2 两相输入加法或减法计数器(UDCNT2)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\[]		特殊功能模 块 []G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
Ⓢ	○ (限于 X)	—	—	—					—
Ⓓ	—	△ * (限于 C)	—	—					—
n	△ *	△ *	△ *	○					—

*:不能使用局部软元件和为个别程序设置的文件寄存器



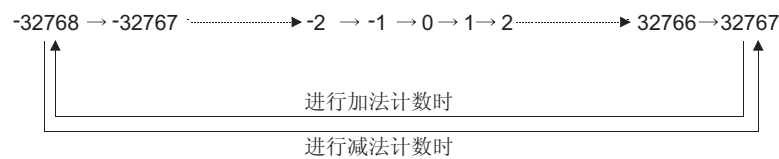
[设定数据]

设定数据	含义	数据类型
Ⓢ	● 用于计数输入的输入数字: Ⓢ+0 (A 相脉冲)	位
	● 用于计数输入的输入数字: Ⓢ+1 (B 相脉冲)	
Ⓓ	● 将根据 UDCNT2 指令进行计数的计数器号	软元件名称
n	● 设定值	BIN 16 位

[功能]

- (1) Ⓓ指定的计数器的当前值根据Ⓢ指定的输入状态(A 相脉冲)和Ⓢ+1 指定的输入的状态(B 相脉冲)实现更新。
- (2) 按以下方式确定计数的方向:
 - 当Ⓢ为 ON 时, 如果Ⓢ+1 从 OFF 变为 ON, 则执行加法计数运算(计数器的当前值增加)。
 - 当Ⓢ为 ON 时, 如果Ⓢ+1 从 ON 变为 OFF, 则执行减法计数运算(计数器的当前值减少)。
 - 如果Ⓢ为 OFF, 则不执行计数运算。
- (3) 计数处理的执行方式描述如下:
 - 当进行加法计数时, 如果当前值变得与 n 指定的设定值相同, 则Ⓓ指定的计数器触点变为 ON。然而, 即使是当Ⓓ指定的计数器触点变为 ON 时, 对当前值的计数还将继续。(参见程序示例(1))
 - 当进行减法计数时, 如果当前值变得比设定值少 1, 则Ⓓ指定的计数器触点变为 OFF。(参见程序示例(1))
 - Ⓓ指定的计数器是环计数器。
当前值为 32767 时, 如果正进行加法计数, 则当前值将变为-32768; 另外, 当前值为-32768 时, 如果正进行减法计数, 则当前值将变为 32767。

基于当前值进行的计数处理如下所示：



- (4) 当计数命令从 OFF 变为 ON 时，基于 UDCNT2 指令的计数处理开始计数；当计数命令从 ON 变为 OFF 时，中止计数。当计数命令再一次从 OFF 变为 ON 时，从计数中止时的值重新开始计数。
- (5) RST 指令清除①指定的计数器的当前值并将触点置为 OFF。

要点

(1) UDCNT2 指令将参数软元件数据注册到 CPU 模块的工作区，而实际的计数运算被处理为系统中断。

(当命令输入置为 OFF 或者当 CPU 模块置为 STOP 然后又置为 RUN 时，将清除注册到 CPU 模块工作区的软元件数据)。

因此，可计数的脉冲的 ON 和 OFF 时间必须比 CPU 模块的中断间隔长。

CPU 模块的中断间隔如下所示。

CPU 模块类型名	中断间隔
高性能型 QCPU，过程控制 CPU	1ms
QnACPU	5ms

(2) 在基于 UDCNT2 指令的一次计数期间(当命令输入为 ON 时)，设定值不能更改。如果要更改设定值，请首先将命令输入置为 OFF。

(3) 其它指令不能使用 UDCNT2 指令指定的计数器。如果这些计数器为其它指令所使用，则它们不能返回精确的计数值。

(4) 在所有正执行的程序内部，UDCNT2 指令最多可以使用 5 次。

第六个以及以后的 UDCNT2 指令不会得到处理。

[运行错误]

- (1) 不存在与 UDCNT2 指令相关的运行错误。

[程序示例]

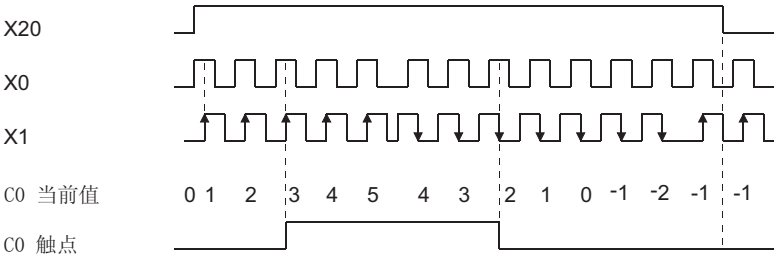
- (1) 在 X20 变为 ON 后，以下程序根据 X0 和 X1 的状态并按照 C0 (加法和减法计数器) 的指示执行一次计数运算。

[梯形图模式]

[列表模式]

步	指令	软元件
0	LD	X20
1	UDCNT2	X0 C0 K3
5	END	

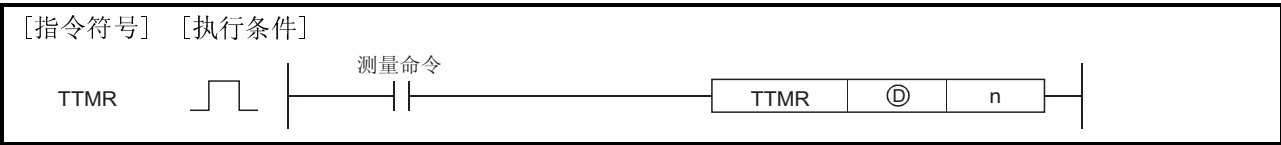
[运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

6.8.3 教学计时器 (TTMR)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J□□□□		特殊功能模 块 U□□□□□	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
①	—	○		—				—	
n	○	○		○				—	



[设定数据]

设定数据	含义	数据类型
Ⓓ	● Ⓓ+0: 存储测量值的软元件	BIN 16 位
	● Ⓓ+1: 为 CPU 模块系统所使用	
n	● 测量值的乘数	

[功能]

- (1) 对测量命令为 ON 的时间以秒为单位进行测量，然后给它乘以 n 指定的乘数，并将结果存储在 Ⓓ指定的软元件中。
- (2) 当测量命令从 OFF 变为 ON 时，会将Ⓓ或者Ⓓ+1 指定的软元件清除。
- (3) n 可以指定的乘数如下所示。

Ⓔ	乘数
0	1
1	10
2	100

要点
(1) 时间测量在执行 TTMR 指令的时候进行。如果使用 JMP 或者类似指令跳转到 TTMR 指令，那么要获得精确的测量值是不可能的。
(2) 在执行 TTMR 指令期间不要更改 n 指定的乘数。更改该乘数会导致返回值不精确。
(3) TTMR 指令也可用于低速执行类型的程序中。
(4) 因为Ⓔ+1 指定的软元件为 CPU 模块系统所使用，因此用户不应该更改它的值。 如果用户确实更改了这个值，那么存储在Ⓔ指定的软元件中的值再也不会是精确的。

(4) 如果“n”指定的值不在范围 0 到 2 之内，则不会执行任何运算。

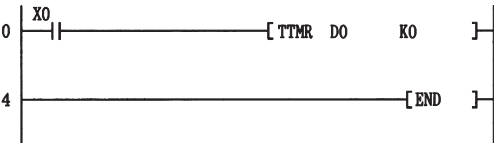
[运行错误]

(1) 不存在与 TTMR 指令相关的运行错误。

[程序示例]

(1) 以下程序将 X0 为 ON 的时间总和存储在 D0 中。

[梯形图模式]



[列表模式]

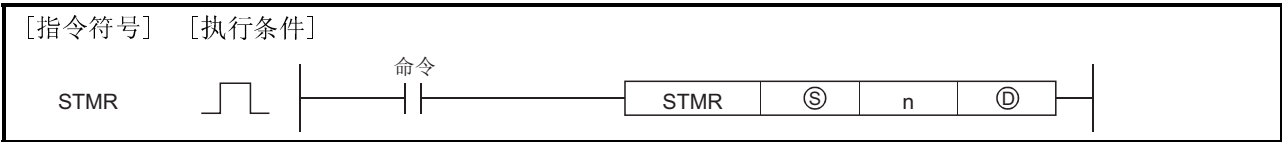
步	指令	软元件
0	LD	X0
1	TTMR	D0
		K0
4	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

6.8.4 特殊功能定时器(STMR)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 []\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
Ⓔ	—	△*	—	—					—
n	○	○	○	○					—
Ⓖ	○	—	—	—					—

※：只能被定时器(T)数据使用



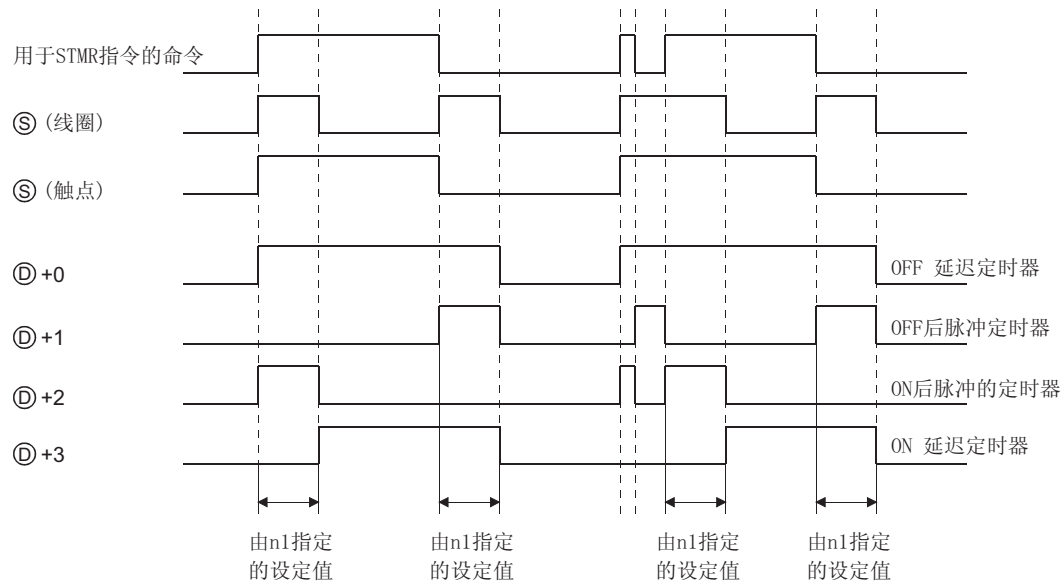
[指令符号]

设定数据	含义	数据类型
Ⓢ	定时器号	字
n	设定数值	BIN 16 位
Ⓓ	● Ⓓ+0: 关延迟定时器输出	位
	● Ⓓ+1: OFF 后脉冲计时器输出	
	● Ⓓ+2: ON 后脉冲计时器输出	
	● Ⓓ+3: ON 延迟定时器输出	

[设定数据]

- (1) STMR 指令使用从Ⓓ指定元件开始的 4 点来执行 4 种类型的定时器输出。
- 关延迟定时器输出 (Ⓓ+0)
在 STMR 指令的上升沿处变为 ON, 并且在命令的下降沿后, n 设定的计数时间结束后变为 OFF。
 - OFF 后脉冲定时器输出 (Ⓓ+1)
在 STMR 指令的下降沿处变为 ON, 并且在 n 指定的时间结束后变为 OFF。
 - ON 后脉冲定时器输出 (Ⓓ+2)
在 STMR 指令的上升沿处变为 ON, 并且在 n 指定的时间过去后或在 STMR 指令的命令变为 OFF 时变为 OFF。
 - 开延迟定时器输出 (Ⓓ+3)
在定时器线圈的下降沿处变为 ON, 并且在 SRMR 指令命令下降沿后, n 设定的计数时间结束后变 OFF。

- (2) Ⓔ指定的定时器线圈在 STMR 指令命令的上升沿处变为 ON, 并开始当前值的测量。
- 当定时器线圈测量到由 n 指定的设定值的点后, 进入时间到达状态并且变为 OFF。
 - 如果 SRMR 指令在定时器线圈到达设定时间之前变为 OFF, 它将保持 ON 状态。这时定时器测量暂停。当 STMR 指令命令再次变为 ON 时, 当前值清 0 并且重新开始测量。
- (3) 定时器触点在 STMR 指令命令的上升沿处变为 ON, 并且在下降沿到达以后, 定时器线圈在 STMR 指令命令的下降沿处变为 OFF。定时器触点只能被 CPU 模块系统使用, 用户不能使用。



- (4) STMR 指令指定的定时器的当前值的测量是在 STMR 指令执行时进行的。
如果 STMR 指令由于 JMP 或类似指令而跳转, 它将不可能得到正确的测量。
- (5) 用于Ⓓ指定的定时器的测量单元与低速定时器是一样的。
- (6) n 值可设定在 1 到 32767 之间。
- (7) Ⓔ指定的定时器不能被 OUT 指令使用。
如果 STMR 和 OUT 指令使用相同的定时器号, 则不能正确运行。

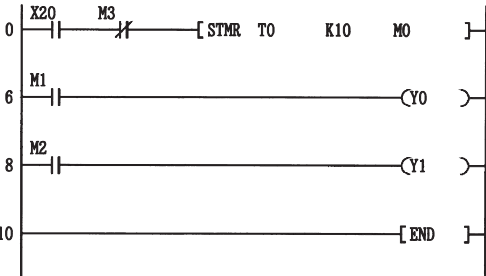
[运行错误]

- (1) 没有与 STMR 指令有关的运行错误。

[程序示例]

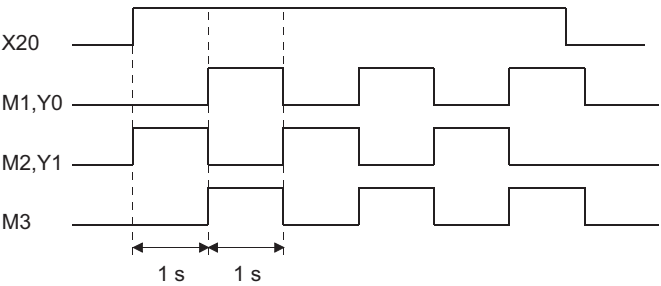
(1) 当 X20 变 ON 时, 下述程序每间隔一秒交替将 Y0 和 Y1 变 ON 和 OFF (闪烁)。(使用 100ms 定时器)

[梯形图模式]



[列表模式]

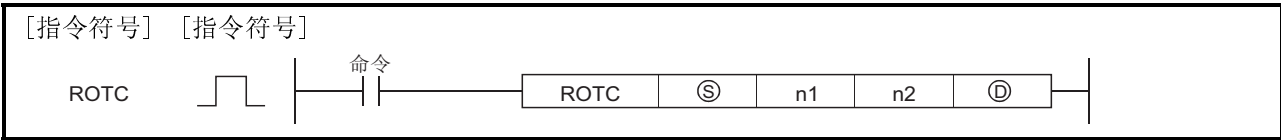
步	指令	软元件
0	LD	X20
1	ANI	M3
2	STMR	T0
		K10
		M0
6	LD	M1
7	OUT	Y0
8	LD	M2
9	OUT	Y1
10	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

6.8.5 旋转台就近路径旋转控制(ROTC)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J□□□		特殊功能模 块 U□\G□	变址寄存器 Zn	常量	其它
	位	字		位	字				
⑤	—	○			—				—
n	○	○			○				—
n2	○	○			○				—
⑥	—	—			—				—



[设定数据]

设定数据	含义	数据类型
Ⓢ	● Ⓢ+0: 测量表(用于系统使用)	BIN 16 位
	● Ⓢ+1: 调用站号	
	● Ⓢ+2: 调用项目号	
n1	● 旋转台的分区数(从 2 到 32767)	
n2	● 低速部分数(值从 0 到 n1)	位
Ⓓ	● Ⓓ+0: 相输入信号	
	● Ⓓ+1: B 相输入信号	
	● Ⓓ+2: 点检测输入信号	
	● Ⓓ+3: 高速向前旋转输出信号 (用于系统使用)	
	● Ⓓ+4: 低速向前旋转输出信号 (用于系统使用)	
	● Ⓓ+5: 停止输出信号 (用于系统使用)	
	● Ⓓ+6: 高速反转输出信号 (用于系统使用)	
	● Ⓓ+7: 低速反转输出信号 (用于系统使用)	

[运算]

- (1) 为了将号码被Ⓢ+2 指定的项目移走或存放在分区数与 n1 指定的值相同的旋转台上, 此控制功能允许旋转台的就近路径旋转到Ⓢ+1 指定的站号位置上。
- (2) 项目号和站号是以逆时针旋转方向分配的项目来控制的。
- (3) 系统使用Ⓢ+0 作为计数器来记录哪个项目号从站号 0 的哪个号开始记数。
不要重写顺控程序数据。
当用户重写了此数据时, 将不可能进行精确控制。
- (4) n2 值应该低于 n1 指定的平台分区数。
- (5) Ⓓ和Ⓓ+1 是用来检测旋转台旋转方向是向前或向后的 A 相或 B 相输入信号。

当 A 相脉冲为 ON, B 相脉冲是在上升沿或下降沿决定了旋转方向。

- 当 B 相为上升沿:向前旋转(顺时针方向旋转)
- 当 B 相为下降沿:向后旋转(逆时针方向旋转)

- (6) 当项目号 0 到达 0 号站时, $\textcircled{D}+2$ 是状态变为 ON 时的 0 点检测输出信号。
当 $\textcircled{D}+2$ 指定的软元件状态变为 ON 并且 ROTC 指令被执行时, $\textcircled{S}+0$ 被清零。
最好先执行清零运行命令, 然后用 ROTC 指令开始就近路径旋转。
- (7) $\textcircled{D}+3$ 到 $\textcircled{D}+7$ 的数据包括用来控制平台运行的输出信号。
 $\textcircled{D}+3$ 到 $\textcircled{D}+7$ 中的一个元件的输出信号将变为 ON 来响应 ROTC 指令的执行结果。
- (8) 如果在 ROTC 指令之前的运行结果变为 OFF, $\textcircled{D}+3$ 到 $\textcircled{D}+7$ 的所有信号都将为 OFF, 不执行就近路径旋转控制运行。
- (9) ROTC 指令在所有程序中只能执行一次。如果多次使用此指令将导致错误运行。
- (10) 当 $\textcircled{S}+0$ 到 $\textcircled{S}+2$, 的值或 n2 的值大于 n1, 不执行此操作。

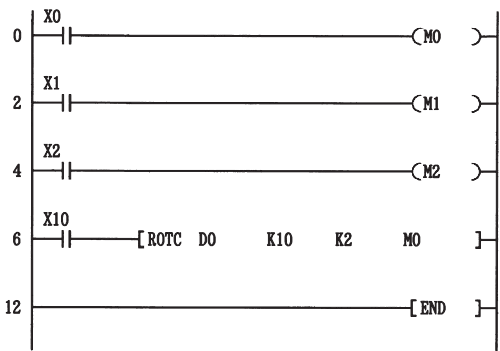
[运行错误]

- (1) 没有与 ROTC 指令相关的错误。

[程序示例]

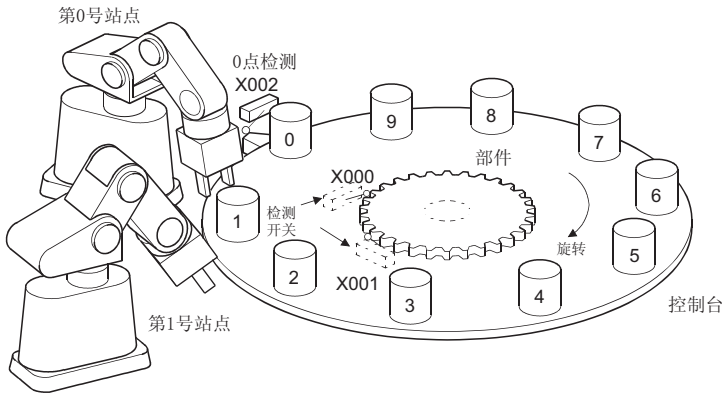
- (1) 下述程序将在 D2 部分的项目存放在旋转台上 D1 部分站, 前面和后面的两部分决定了在旋转台低速旋转时的马达的旋转方向和控制速度。

[梯形图模式]



[列表模式]

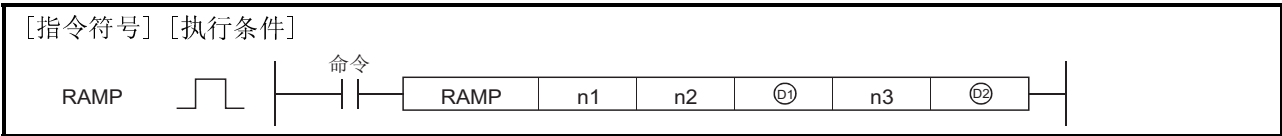
步	指令	软元件
0	LD	X0
1	OUT	M0
2	LD	X1
3	OUT	M1
4	LD	X2
5	OUT	M2
6	LD	X10
7	ROTC	D0 K10 K2 M0
12	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

6.8.6 斜坡信号(RAMP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
n1	○				○			○	—
n2	○				○			○	—
①	○				○			—	—
n3	○				○			○	—
②	○				—			—	—



[设定数据]

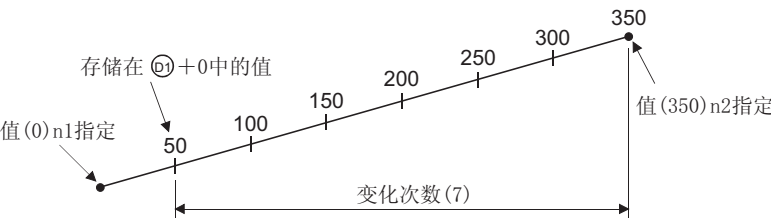
设定数据	含义	数据类型
n1	● 最终值	BIN 16 位
n2	● 初始值	
①	● ①+0: 当前值	
	● ①+1: 执行次数 (系统使用)	
n3	● 移动次数	位
②	● ②+0: 完成软元件	
	● ②+1: 用于在完成时存储数据的选择位	

[功能]

(1) 将值存储在①+1 中，该值按照 n3 指定的移动次数将 n1 指定的值线性变化到 n2 指定的值。存储在①+1 中的值是通过下列表达式在每个扫描周期计算出来的。

$$\frac{\{(n2指定的值) - (n1指定的值)\}}{(移动次数)} \times (执行次数)$$

如下所示，在 6 个扫描周期中从 0 变化到 350。



当在一次变化中除不尽时，要通过 n3 指定的移动次数得到 n2 指定的值，需要做一些补偿。因此，可能不能生成线性的斜坡。

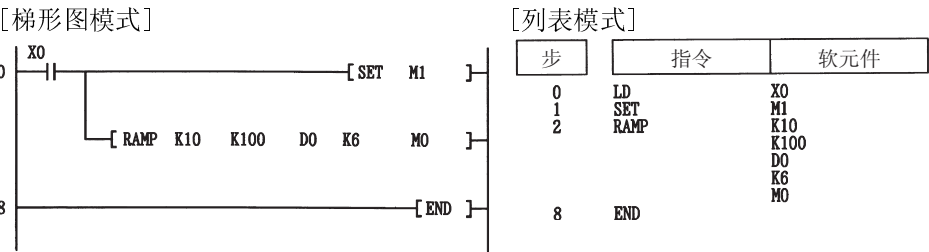
- (2) 对 n3 而言，它指定将数据从 n1 移动到 n2 所要求的扫描次数。
当 n3=0 时，不执行任何处理。
- (3) 系统使用③+1 来存储指令已经执行的次数。
- (4) 当成功地完成到最终值的移动时，③+0 指定的完成软元件将变为 ON。
完成软元件的 ON 或 OFF 状态和③+0 的内容由③+1 指定的软元件的 ON 或 OFF 状态决定。
- 当③+1 为 OFF 时，③+0 将在下一次扫描时变为 OFF，并且 RAMP 指令将从③+0 中的当前值开始进行一次新的移动运算。
 - 当③+1 为 ON 时，③+0 将保持为 ON，③+0 的内容也保持不变。
- (5) 在该指令的执行期间，当命令变为 OFF 时，③+0 的内容不会因此而改变。
当该命令再次变为 ON 时，RAMP 指令将从③+0 中的当前值开始进行一次新的移动。
- (6) 在③+0 指定的完成软元件变为 ON 之前不要更改 n1 和 n2 指定的值。
因为在每次扫描期间都使用相同的表达式来计算存储在③+1 中的值，所以，更改 n1 或 n2 的值可能导致突然的变化。

[运行错误]

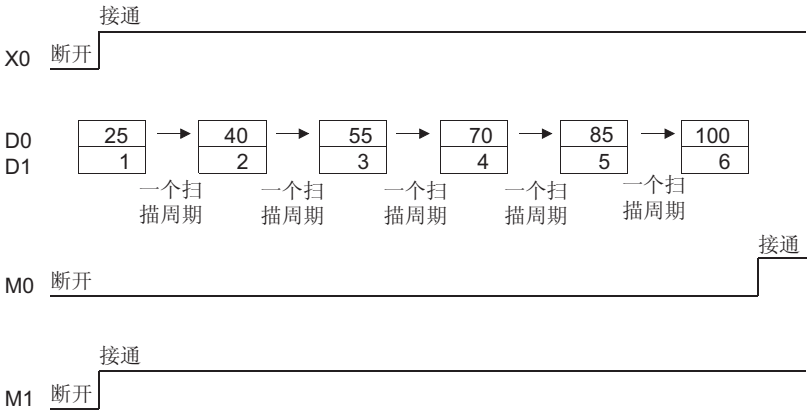
- (1) 不存在与 RAMP 指令相关的运行错误。

[程序示例]

- (1) 以下程序在总共 6 次的扫描中将 D0 中的内容从 10 变化到 100，并在完成移动时保存 D0 的内容。



[运算]

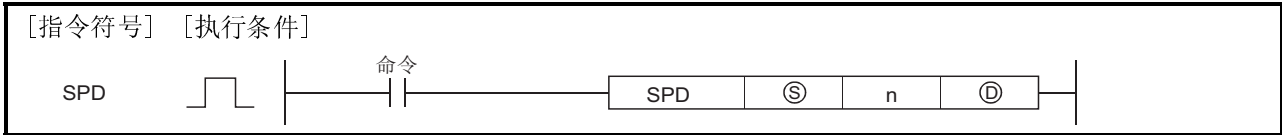


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

6.8.7 脉冲密度测量 (SPD)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量	其它
	位	字		位	字				
⑤	○ (限于 X)	—	—					—	
n	△ *	△ *	○					—	
⑥	—	△ *	—					—	

※: 不能使用局部软元件和为个别程序设置的文件寄存器

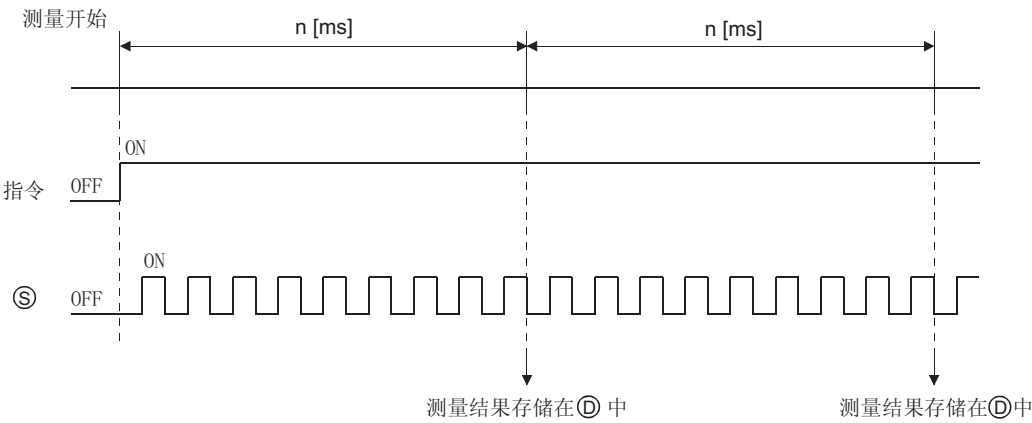


[设定数据]

设定数据	含义	数据类型
Ⓢ	● 脉冲输入	位
n	● 测量时间(单位: ms)	BIN 16 位
Ⓓ	● 存储测量结果的软元件的起始号	

[功能]

(1) 只在 n 指定的时间内对来自Ⓢ指定的软元件的输入进行计数，并将计数结果存储在Ⓓ指定的软元件中。



(2) 如果 SPD 指令引导的测量已经完成，则从 0 开始进行下一次测量。
要中止 SPD 指令引导的测量，请将命令变为 OFF。

要点						
<p>(1) SPD 指令将参数软元件中的数据注册到 CPU 模块的工作区，而实际的计数运算在系统中断期间执行。</p> <p>(当命令输入变为 OFF 或者当 CPU 模块变为 STOP 然后又变为 RUN 时，将清除注册到 CPU 模块工作区的软元件数据)。</p> <p>因此，可计数的脉冲的 ON 和 OFF 时间必须比 CPU 模块的中断间隔长。</p> <p>CPU 模块的中断间隔如下所示。</p> <table border="1"><thead><tr><th>CPU 模块类型名</th><th>中断间隔</th></tr></thead><tbody><tr><td>高性能型 QCPU, 过程控制 CPU</td><td>1ms</td></tr><tr><td>QnACPU</td><td>5ms</td></tr></tbody></table>	CPU 模块类型名	中断间隔	高性能型 QCPU, 过程控制 CPU	1ms	QnACPU	5ms
CPU 模块类型名	中断间隔					
高性能型 QCPU, 过程控制 CPU	1ms					
QnACPU	5ms					
<p>(2)</p> <ul style="list-style-type: none">● 使用高性能型 QCPU 或过程控制 CPU 的时候： 当 n=0 时，不对指令进行处理。● 使用 QnACPU 的时候： 当 n=0 或者 n 不是 5 的倍数时，不对指令进行处理。						
<p>(3) 在所有正执行的程序内部，SPD 指令最多可以使用 6 次。</p> <p>第七个以及以后的 SPD 指令不会得到处理。</p>						

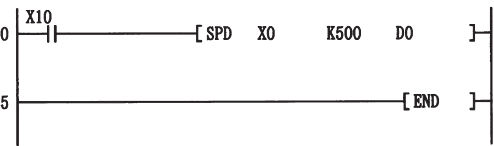
[运行错误]

(1) 不存在与 SPD 指令相关的运行错误。

[程序示例]

(1) 当 X10 变为 ON 时，以下程序在 500ms 的时间内对输入到 X0 的脉冲进行测量，并将结果存储在 D0 中。

[梯形图模式]



[列表模式]

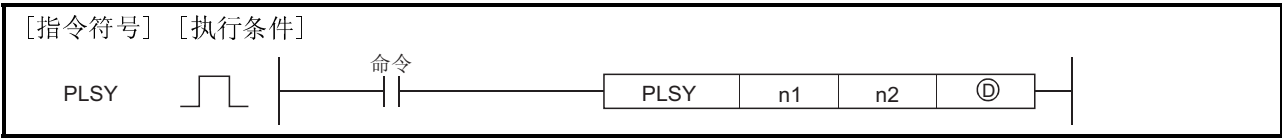
步	指令	软元件
0	LD	X10
1	SPD	X0 K500 D0
5	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

6.8.8 固定周期脉冲输出 (PLSY)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]G[][]	变址寄存器 Zn	常量	其它
	位	字		位	字				
n1	○				○				—
n2	○				○				—
⑤	△ *				—				—

※:只能使用输出 (Y)



[设定数据]

设定数据	含义	数据类型
n1	● 设置频率的软元件号	BIN 16 位
n2	● 设置输出次数的软元件号	
Ⓓ	● 执行脉冲输出的软元件号	位

[功能]

- (1) 以 n1 指定的频率将脉冲输出到带有 D1 指定的输出信号 (Y) 的输出模块，其中输出次数由 n2 指定。
- (2) n1 可以指定的频率范围是 1 到 100Hz。
如果 n1 超出了范围 1 到 100Hz，则 PLSY 指令将不会得到执行。
- (3) n2 可以指定的输出次数的范围是 1 到 65535 (0000H 到 0FFFFH)。
- (4) 只有一个与输出模块相对应的输出号可以指定给Ⓓ中的脉冲输出。
- (5) 脉冲输出开始于 PLSY 指令命令的上升沿。在脉冲输出期间，不要将 PLSY 指令命令变为 OFF。
PLSY 指令命令变为 OFF 时，脉冲输出将中止。

要点

- (1) PLSY 指令将参数软元件数据注册到 CPU 模块的工作区，而实际的输出运算在系统中断期间执行。
(当命令输入变为 OFF 或者当 CPU 模块变为 STOP 然后又变为 RUN 时，将清除注册到 CPU 模块工作区的软元件数据)。
因此，可计数的脉冲的 ON 和 OFF 时间必须比 CPU 模块的中断间隔长。
CPU 模块的中断间隔如下所示。

CPU 模块类型名	中断间隔
高性能型 QCPU, 过程控制 CPU	1ms
QnACPU	5ms

- (2) 在通过 PLSY 指令输出脉冲的过程中，不要更改 PLSY 指令的参数 (在命令输入变为 ON 时。) 在更改参数之前请先将命令输入变为 OFF。
(3) 因此，在 CPU 模块执行的整个程序中，PLSY 指令只能使用一次。

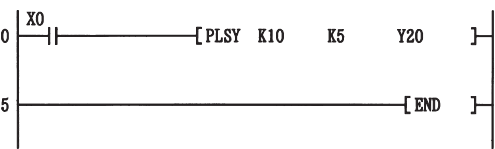
[运行错误]

- (1) 不存在与 PLSY 指令相关的运行错误。

[程序示例]

- (1) 当 X0 变为 ON 时，以下程序将一个 10Hz 脉冲输出到 Y20，输出次数为 5。

[梯形图模式]



[列表模式]

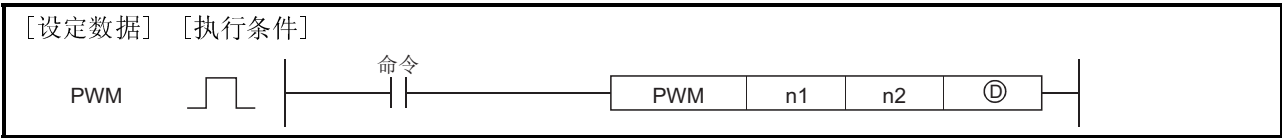
步	指令	软元件
0	LD	X0
1	PLSY	K10 K5 Y20
5	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

6.8.9 脉冲宽度调制(PWM)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J K		特殊功能模 块 U V G	变址寄存器 Zn	常量	其它
	位	字		位	字				
n1	○				○				—
n2	○				○				—
⑤	△ *				—				—

※：只能使用输出(Y)

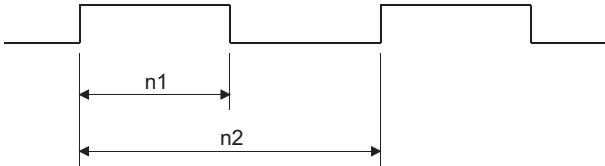


[设定数据]

设定数据	含义	数据类型
n1	● 设置 ON 状态(接通)时间的软元件号	BIN 16 位
n2	● 设置周期的软元件号	
Ⓓ	● 执行脉冲输出的软元件号	位

[功能]

(1) 在 n1 指定的状态为 ON(接通)的时间中，将脉冲(周期为 n2 所设定)输出到Ⓓ指定的输出模块。



(2) n1 和 n2 的设置范围如下所示：

CPU 模块类型名	n1*和 n2 的设定范围[ms]
高性能型 QCPU, 过程控制 CPU	1 到 65535 (0001 _H 到 0FFFF _H)
QnACPU	5 到 65535 (0005 _H 到 0FFFF _H)

※:指定给 n1 的值应该小于或者等于指定给 n2 的值

[运行错误]

(1) 不存在与 PLSY 指令相关的运行错误。

要点

(1) PWM 指令将指定的软元件数据注册到 CPU 模块的工作区。实际的输出运算被 CPU 模块处理为中断。

(当命令输入变为 OFF 或者当 CPU 模块变为 STOP 然后又变为 RUN 时，将清除注册到 CPU 模块工作区的软元件数据)。

因此，可计数的脉冲的 ON 和 OFF 时间必须比 CPU 模块的中断间隔长。

CPU 模块的中断间隔如下所示。

CPU 模块类型名	n1, n2 的中断间隔
高性能型 QCPU, 过程控制 CPU	1ms
QnACPU	5ms

因此，在 CPU 模块执行的所有程序中，PWM 指令只能使用一次。

(2) 在下列情况下指令不会得到执行：

- 当 n1 和 n2 都为 0 时
- 当 n1 和 n2 不都是 5 的倍数时(只有当使用 QnACPU 的时候)
- 当 $n2 \geq n1$ 时

(3) 在通过 PWM 指令输出脉冲的过程中，不要更改 PWM 指令的参数(在命令输入变为 ON 时)。

在更改参数之前请先将命令输入变为 OFF。

[程序示例]

(1) 当 X0 变为 ON 时，以下程序将一个 100ms 的脉冲输出到 Y20，频率为每秒钟一次。

[梯形图模式]

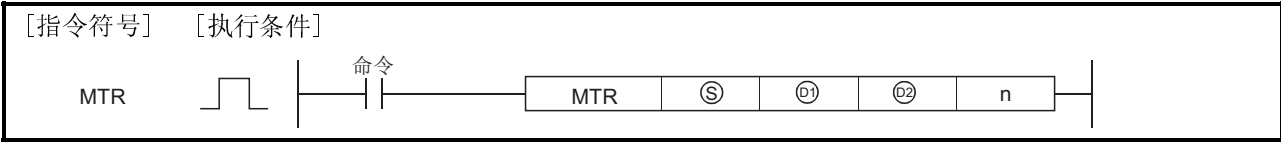
[列表模式]

步	指令	软元件
0	LD	X0
1	PWM	K100 K1000 Y20
5	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

6.8.10 矩阵输入(MTR)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\K[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量	其它
	位	字		位	字				
⑤	○	—						—	
⑥1	○	—						—	
⑥2	○	—						—	
n	○	○						—	



[设定数据]

设定数据	含义	数据类型
⑤	● 起始输入软元件	位
⑥1	● 起始输出软元件	
⑥2	● 存储矩阵输入数据的起始号软元件	
n	● 输入行数	BIN 16 位

[功能]

- (1) 连续读取从⑤指定的输入地址起始的 16×n 点数据(连续读取 n 行 16 点数据)，然后将从该运算中接收到的数据存储在⑥2指定的软元件起始的软元件中。
- (2) 在一个扫描周期中可以接收一行(16 点)。
- (3) 从第一行到第 n 行的数据接收的过程是循序渐进的重复过程。
- (4) 在从⑥2指定的软元件起始的软元件中，第一个 16 点(第 1 到 16 点)存储第一行数据，下一个 16 点存储第二行数据。因此，从⑥2指定的软元件起始的 16×n 点空间为 MTR 指令所占据。
- (5) ⑥2是用于选择将被读取的行的输出，且系统自动将它变为 ON 和 OFF。
它使用从 D1 指定的软元件起始的 n 点。
- (6) 只有可被 16 整除的软元件号才可以指定给⑤、⑥1和⑥2。
- (7) n2 的值不在范围 2 到 8 之内。(错误代码: 4100)

- (8) 在下列情况下不执行任何处理：
- ⑤、⑥或⑦指定的软元件号不能被 16 整除。
 - ⑤指定的软元件超出了实际的输入范围。
 - ⑥指定的软元件超出了实际的输出范围。
 - 从⑦指定的软元件起始的 $16 \times n$ 点空间超出了相关软元件范围。
 - $n2$ 的值不在范围 2 到 8 之内。

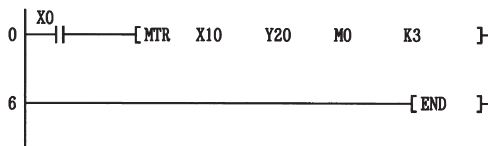
[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- ⑤指定了除输入(X)之外的软元件。(错误代码：4101)
 - ⑥指定了除输出(Y)之外的软元件。(错误代码：4101)

[程序示例]

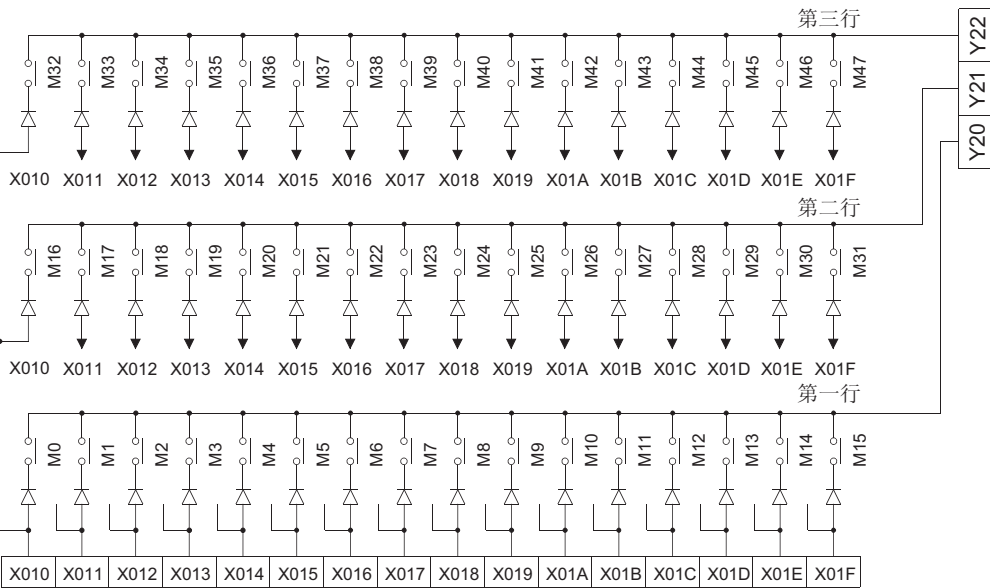
- (1) 当 X0 变为 ON 时，以下程序接收从 X10 起始的 16 点 \times 3 行数据，并将它存储在从 M30 起始的软元件中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	MTR	X10 Y20 M0 K3
6	END	



[注意]

- (1) 注意，MTR 指令对实际的输入和输出直接进行操作。
已被 MTR 指令变为 ON 的输出⑥不会在 MTR 命令变为 OFF 时变为 OFF。指定的输出⑥在顺序程序中变为 OFF。
- (2) MTR 指令的执行间隔必须比输入和输出模块响应时间的总和长。
如果设定的间隔比以上表示的值短，则不能正确读取输入。
如果顺序程序中的扫描时间太短，则选择恒定扫描并将其扫描时间设变得比响应时间的总和长。

备忘录

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

7. 应用指令

提供了以下应用指令：

指令	含义	参考章节
逻辑运算指令	逻辑加、逻辑乘等逻辑运算	7.1 节
循环移动指令	指定数据的循环移动	7.2 节
移位指令	指定数据的移位	7.3 节
位处理指令	置位和复位位数据;位提取	7.4 节
数据处理指令	数据查找, 解码, 编码等数据处理	7.5 节
结构化程序指令	重复运算, 子程序调用和梯形图单元中的变址修改	7.6 节
数据表运算指令	读/写 FIFO 表	7.7 节
缓冲存储器访问指令	读/写特殊功能模块的缓冲存储器	7.8 节
显示指令	将字符代码输出到外部元件, 并显示在显示单元上	7.9 节
调试与故障诊断指令	检查、状态锁存、采样追踪和程序跟踪	7.10 节
字符串处理指令	字符串 (ASCII 码数据) 处理	7.11 节
特殊功能指令	BCD 型实数处理; 浮动小数点类型实数处理	7.12 节
数据控制指令	基于输入数据范围检测的输出值控制	7.13 节
文件寄存器切换指令	设置文件寄存器; 切换块号码	7.14 节
时钟指令	读/写时钟数据	7.15 节
外围设备指令	显示外围设备消息; 键输入	7.16 节
程序控制指令	切换程序执行条件的指令	7.17 节
其它指令	与以上各类不符的指令, 比如看门狗计时器复位指令和计时时钟指令	7.18 节

7.1 逻辑运算指令

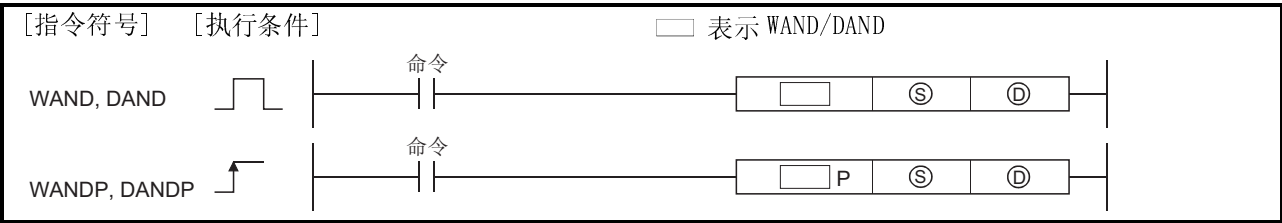
(1) 逻辑运算指令包括逻辑和逻辑乘等, 它们在 1 位单元中进行处理, 讨论如下。

分类	处理详细信息	运算公式	示例		
			A	B	Y
逻辑乘(AND)	只有当输入 A 和输入 B 都为 1 时才变为 1 否则, 为 0	$Y=A \cdot B$	0	0	0
			0	1	0
			1	0	0
			1	1	1
逻辑和(OR)	只有当输入 A 和输入 B 都为 0 时才变为 0 否则, 为 1	$Y=A+B$	0	0	0
			0	1	1
			1	0	1
			1	1	1
异或 OR(XOR)	如果输入 A 和输入 B 相等, 则变为 0, 否则 为 1	$Y=\overline{A} \cdot B+A \cdot \overline{B}$	0	0	0
			0	1	1
			1	0	1
			1	1	0
异或非(XNR)	如果输入 A 和输入 B 相等, 则变为 1, 否则 为 0	$Y=(\overline{A+B}) (A+B)$	0	0	1
			0	1	0
			1	0	0
			1	1	1

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.1.1 16 位和 32 位数据的逻辑乘 (WAND, WANDP, DAND, DANDP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	



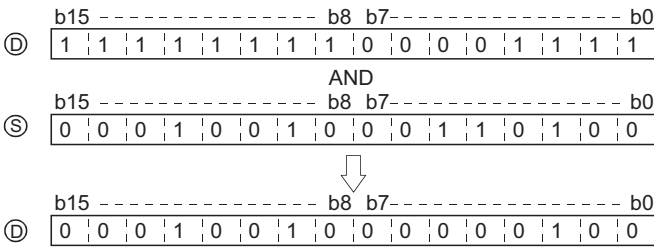
[设定数据]

设定数据	含义	数据类型
⑤	生成逻辑乘的数据, 或者存储这些数据的软元件的起始号	BIN 16/32 位
⑥	存储被执行“逻辑乘”运算数据的软元件的起始号	

[功能]

WAND

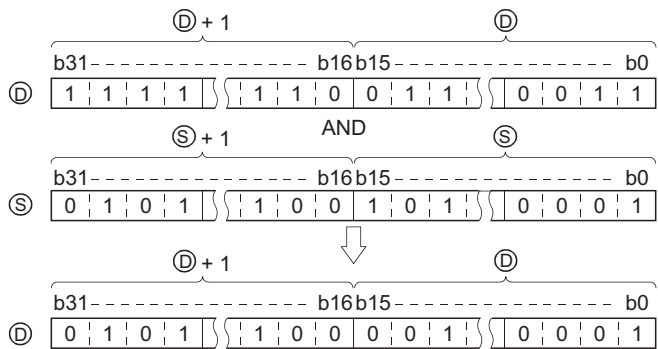
(1) 对⑥指定的软元件中的 16 位数据的每一位与⑤指定的软元件中的 16 位数据的对应位进行逻辑乘运算, 并将结果存储在⑥指定的软元件中。



(2) 对位软元件而言, 高于指定位数的数字位在该运算中被处理为 0。

DAND

(1) 对⑥指定的软元件中的 32 位数据的每一位与⑤指定的软元件中的 32 位数据的对应位进行逻辑乘运算, 并将结果存储在⑥指定的软元件中。



(2) 对位软元件而言, 高于指定位数的数字位在该运算中被处理为 0。[参见程序示例 (2)]

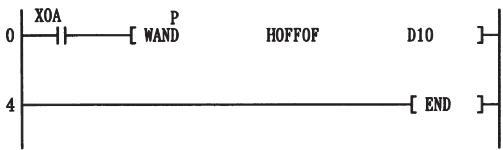
[运行错误]

(1) 不存在与 WAND (P) 或 DAND (P) 指令相关的运行错误。

[程序示例]

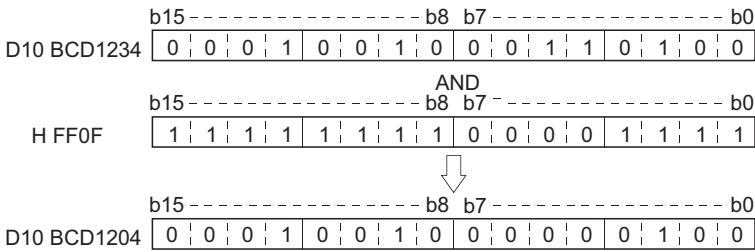
(1) 当 XA 为 ON 时, 以下程序将 D10 中 4 位 BCD 值的十位数 (右起第二位) 上的数字屏蔽为 0。

[梯形图模式]



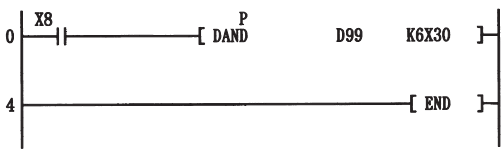
[列表模式]

步	指令	软元件
0	LD	XA
1	WANDP	HOFFOF D10
4	END	



(2) 当 X8 为 ON 时, 以下程序对 D99 到 D100 中的数据 and X30 到 X47 中的 24 位数据执行逻辑乘运算, 并将结果存储在 D99 到 D100 中。

[梯形图模式]

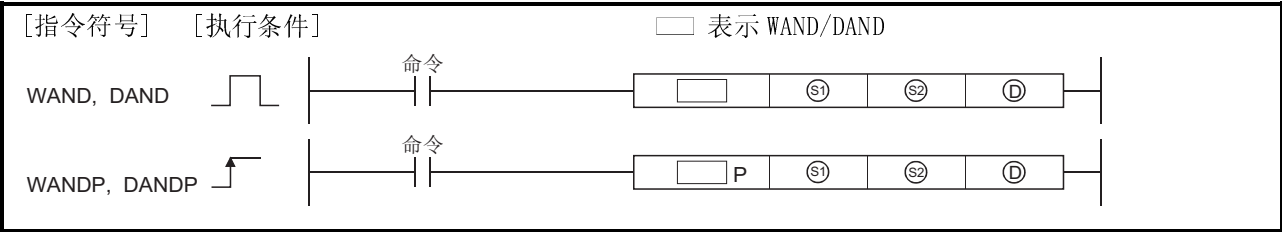


[列表模式]

步	指令	软元件
0	LD	X8
1	DANDP	D99 K6X30
4	END	



设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][]G[][]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				U
Ⓔ1	○							○	—
Ⓔ2	○							○	—
Ⓓ	○							—	—



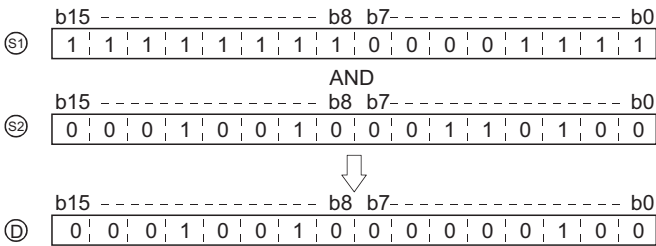
[设定数据]

设定数据	含义	数据类型
Ⓢ1	生成逻辑乘的数据, 或者存储这些数据的软元件的起始号	BIN 16/32 位
Ⓢ2		
Ⓓ	存储逻辑乘运算结果的软元件的起始号	

[功能]

WAND

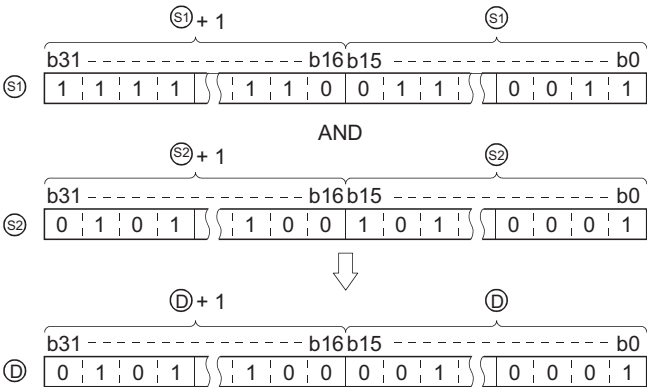
(1) 对Ⓢ1指定的软元件中的 16 位数据的每一位与Ⓢ2指定的软元件中的 16 位数据的对应位进行逻辑乘运算, 并将结果存储在Ⓓ指定的软元件中。



(2) 对位软元件而言, 高于指定位数的数字位在该运算中被处理为 0。[参见程序示例(1)和(2)]

DAND

(1) 对①指定的软元件中的 32 位数据的每一位与②指定的软元件中的 32 位数据的对应位进行逻辑乘运算, 并将结果存储在③指定的软元件中。



(2) 对位软元件而言, 高于指定位数的数字位在该运算中被处理为 0。[参见程序示例 (3)]

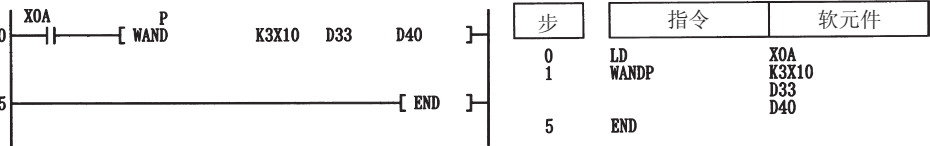
[运行错误]

(1) 不存在与 WAND (P) 或 DAND (P) 指令相关的运行错误。

[程序示例]

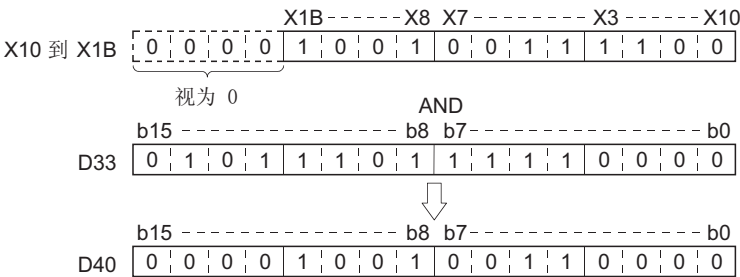
(1) 当 XA 为 ON 时, 以下程序对 X10 到 X1B 中的数据和 D33 中的数据执行逻辑乘运算, 并将结果存储在 D40 中。

[指令符号]

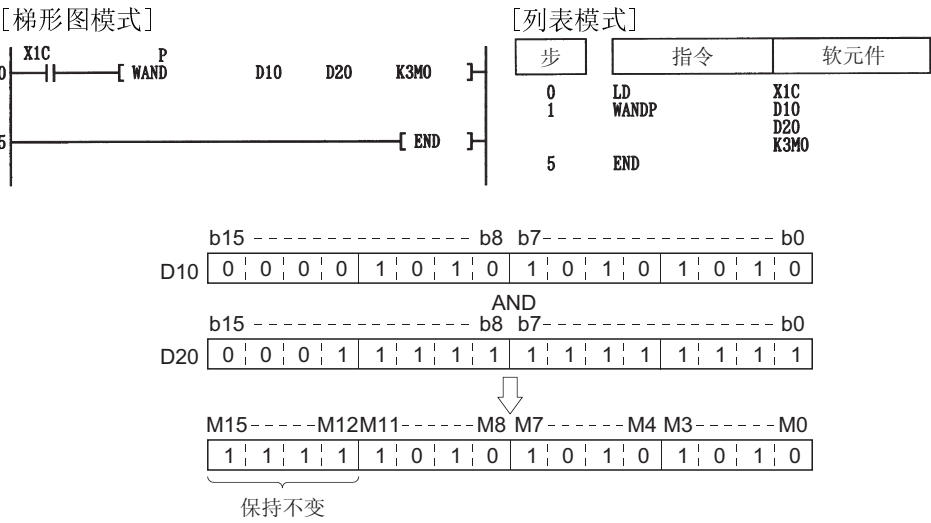


[执行条件]

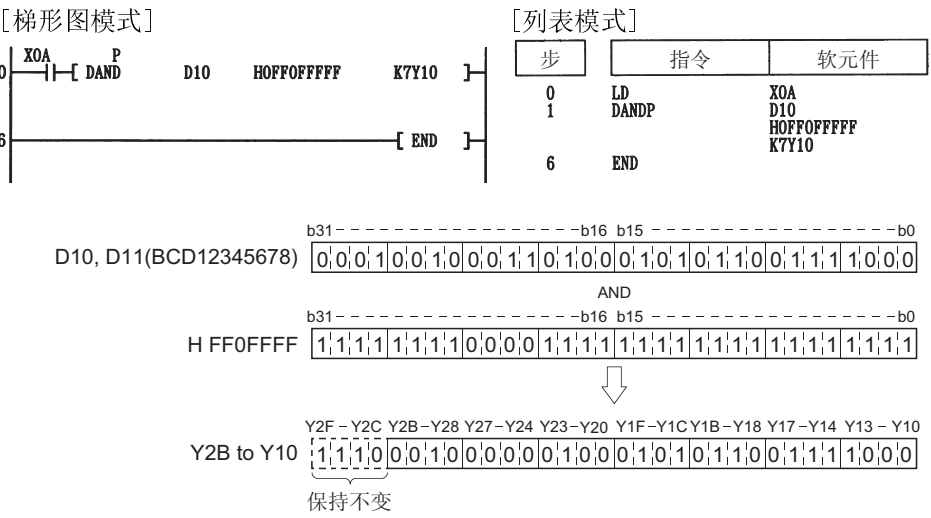
步	指令	软元件
0	LD	X0A
1	WANDP	K3X10 D33 D40
5	END	



(2) 当 X1C 为 ON 时, 以下程序对 D10 中的数据 and D20 中的数据执行逻辑乘运算, 并将结果存储在 M0 到 M11 中。



(3) 当 XA 为 ON 时, 以下程序将 D10, D11 中 8 位 BCD 值十万位 (右起第六位) 上的数字屏蔽为 0, 并将结果输出到 Y10 到 Y2B。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.1.2 块逻辑乘(BKAND, BKANDP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[] []		特殊功能模 块 U[] [] G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
①	—	○			—			—	—
②	—	○			—			○	—
③	—	○			—			—	—
n	○	○			○			○	—

[指令符号]	[执行条件]
BKAND	
BKANDP	

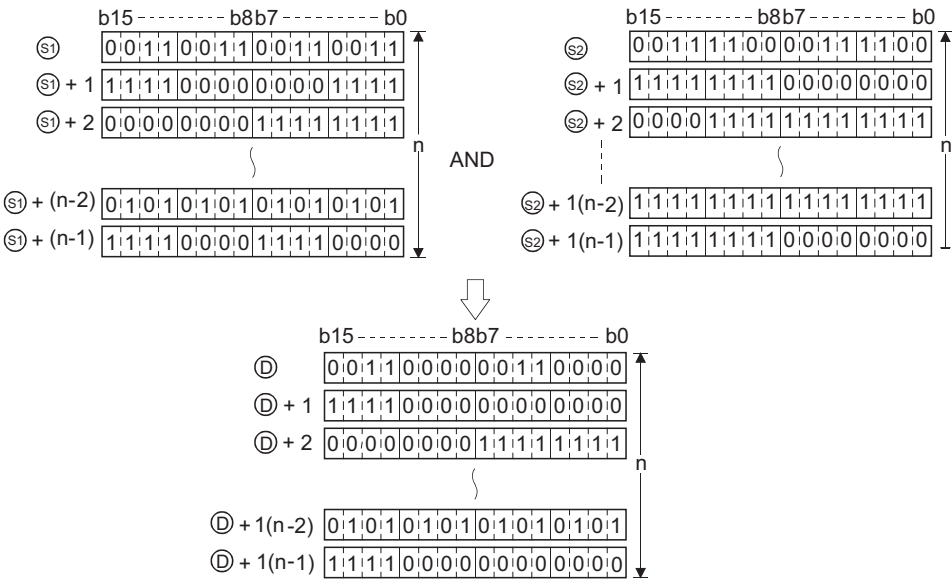
[设定数据]

设定数据	含义	数据类型
①*	存储要用于逻辑乘运算的数据的软元件的起始号	BIN 16 位
②*	逻辑运算数据的第一个数或者存储逻辑运算数据的软元件的起始号	
③*	存储运算结果的软元件的起始号	
n	运算数据的个数	

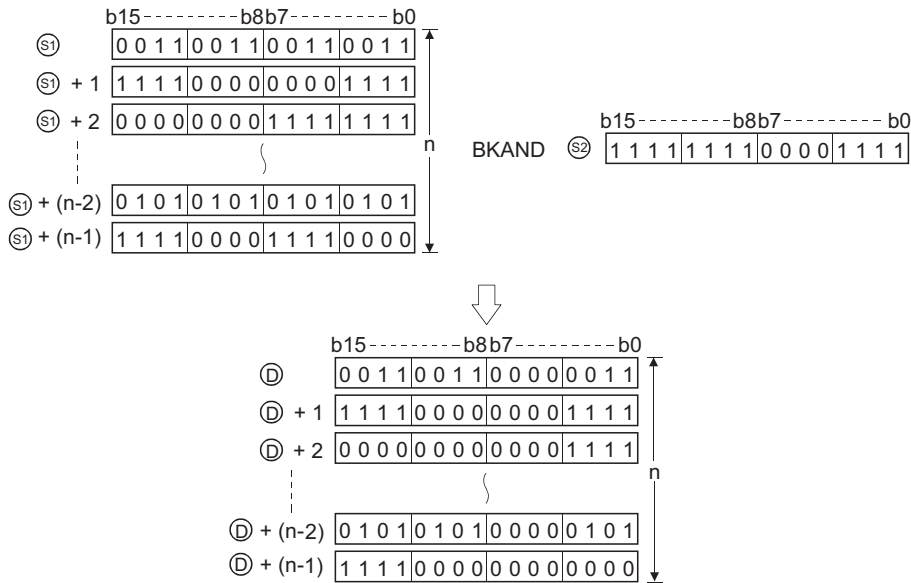
*: 可以给①和③或者②和③指定相同的软元件号。

[功能]

- (1) 对从①指定的软元件开始的 n 点中的数据 and 从②指定的软元件开始的 n 点中的数据执行逻辑乘运算, 并将结果存储在③指定的软元件中。



(2) ②可以指定的常量的范围是-32768 到 32767 (BIN 16 位)。



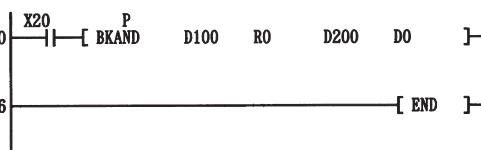
[运行错误]

- (1) 在发生以下运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 从 ③, ②或 ④软元件开始的 n 点范围超出了该软元件的范围。 (错误代码: 4101)
 - 从 ③软元件开始的 n 点软元件范围与从 ④软元件开始的 n 点软元件范围相互重叠。 (指定给 ③和 ④的软元件相同时除外) (错误代码: 4101)
 - 从 ②软元件开始的 n 点软元件范围与从 ④软元件开始的 n 点软元件范围相互重叠。 (指定给 ③和 ④的软元件相同时除外) (错误代码: 4101)
 - 从 ③软元件开始的 n 点软元件范围与从 ②软元件开始的 n 点软元件范围相互重叠。 (错误代码: 4101)

[程序示例]

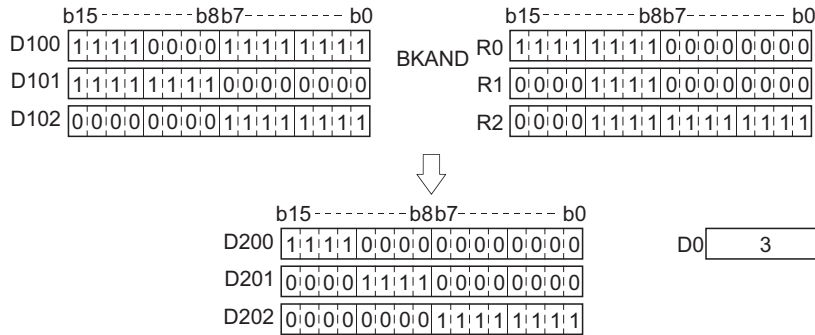
(1) 当 X20 变为 ON 时, 以下程序对与存储在 D100 和 D0 中的值相对应的数据点数和与存储在 R0 和 D0 中的值相对应的数据点数执行逻辑乘运算, 并将结果存储在从 D200 开始的软元件中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X20
1	BKANDP	D100 R0 D200 D0
6	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.1.3 16 位和 32 位数据的逻辑和 (WOR, WORP, DOR, DORP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	

[指令符号]	[执行条件]	□ 表示 WOR/DOR
WOR, DOR		
WORP, DORP		

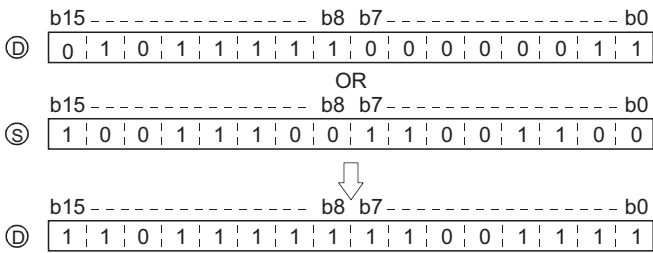
[设定数据]

设定数据	含义	数据类型
⑤	用于执行逻辑和运算的数据, 或者存储该数据的软元件的起始号	BIN 16/32 位
⑥	存储被执行“逻辑和”运算的数据的软元件的起始号	

[功能]

WOR

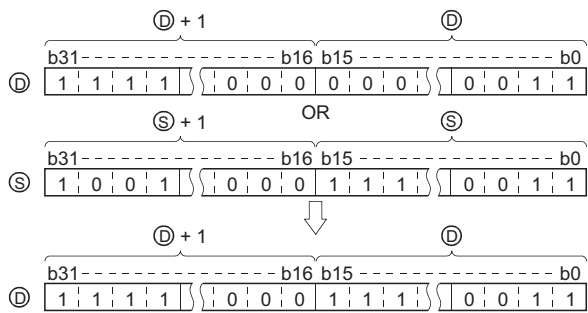
- (1) 对⑥指定的软元件中的 16 位数据的每一位与⑤指定的软元件中的 16 位数据的对应位进行逻辑和运算, 并将结果存储在⑥指定的软元件中。



- (2) 对位软元件而言, 高于指定位的位数在该运算中被处理为 0。

DOR

- (1) 对⑥指定的软元件中的 32 位数据的每一位与⑤指定的软元件中的 32 位数据的对应位进行逻辑和运算, 并将结果存储在⑥指定的软元件中。



(2) 对位软元件而言, 高于指定位的位数在该运算中被处理为 0。

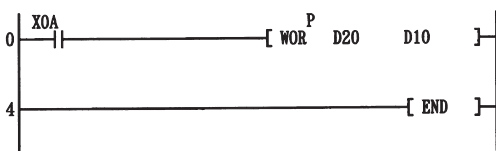
[运行错误]

(1) 不存在与 WOR (P) 或 DOR (P) 指令相关的运行错误。

[程序示例]

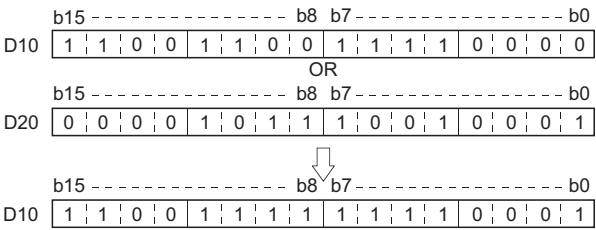
(1) 当 XA 变为 ON 时, 以下程序对 D10 和 D20 中的数据执行逻辑和运算, 并将结果存储在 D10 中。

[梯形图模式]



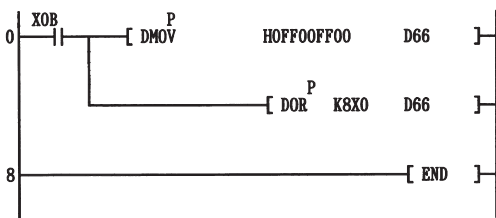
[列表模式]

步	指令	软元件
0	LD	X0A
1	WORP	D20 D10
4	END	



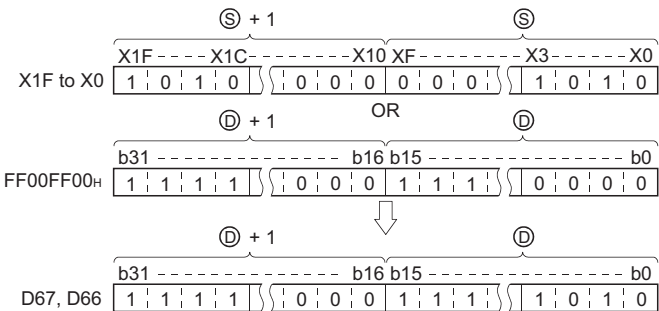
(2) 当 XB 变为 ON 时, 以下程序对 X0 到 X1F 中的 32 位数据和 16 进制值 FF00FF00H 执行逻辑和运算, 并将结果存储在 D66 和 D67 中。

[梯形图模式]

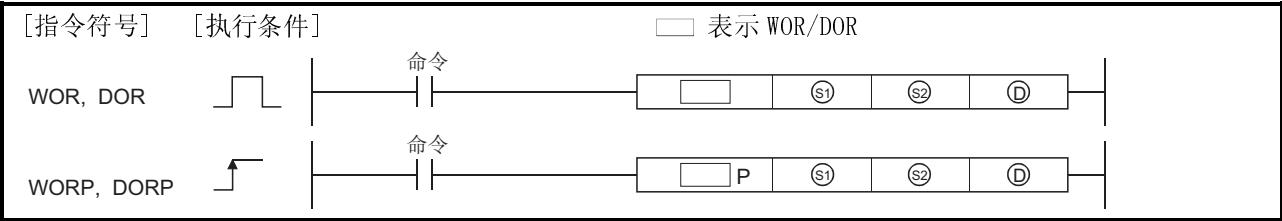


[列表模式]

步	指令	软元件
0	LD	X0B
1	DMOVP	FF00FF00 D66
5	DORP	X0 D66
8	END	



设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J K L		特殊功能模 块 U V G L	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				U
①	○						○	—	
②	○						○	—	
③	○						—	—	



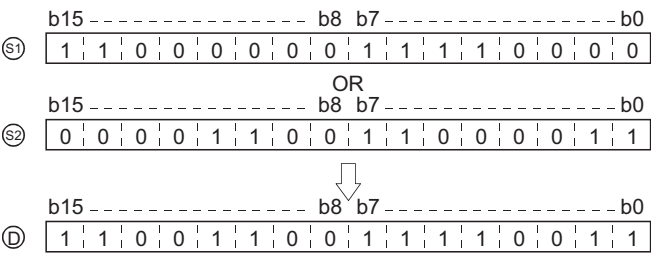
[设定数据]

设定数据	含义	数据类型
①	用于执行逻辑和运算的数据, 或者存储该数据的软元件的起始号	BIN 16/32 位
②		
③	存储逻辑和运算结果的软元件的起始号	

[功能]

WOR

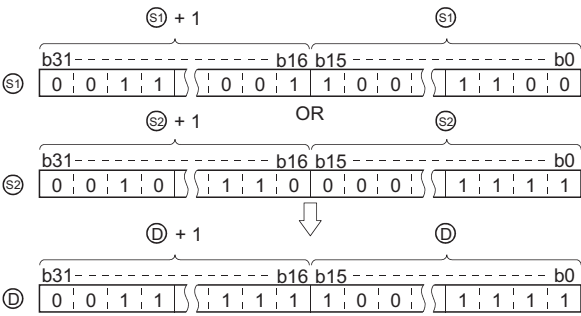
- (1) 对①指定的软元件中的 16 位数据的每一位与②指定的软元件中的 16 位数据的对应位进行逻辑和运算, 并将结果存储在③指定的软元件中。



- (2) 对位软元件而言, 高于指定位的位数在该运算中被处理为 0。[参见程序示例(1)]

DOR

- (1) 对①指定的软元件中的 32 位数据的每一位与②指定的软元件中的 32 位数据的对应位进行逻辑和运算, 并将结果存储在③指定的软元件中。



(2) 对位软元件而言, 高于指定位的位数在该运算中被处理为 0。[参见程序示例(2)]

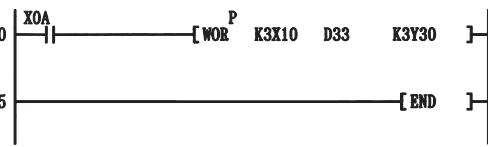
[运行错误]

(1) 不存在与 WOR (P) 或 DOR (P) 指令相关的运行错误。

[程序示例]

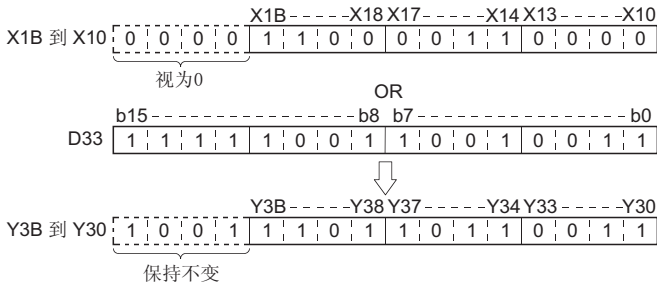
(1) 当 XA 为 ON 时, 以下程序对 X10 到 X1B 中的数据和 D33 中的数据执行逻辑和运算, 并将结果存储在 Y30 到 Y3B 中。

[梯形图模式]



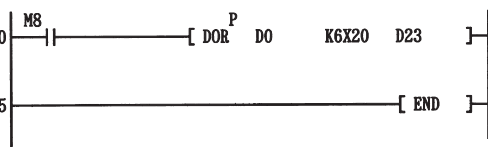
[列表模式]

步	指令	软元件
0	LD	X0A
1	WORP	K3X10 D33 K3Y30
5	END	



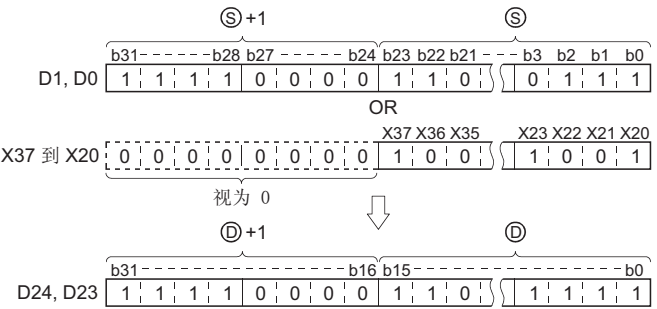
(2) 当 M8 为 ON 时, 以下程序对 D0, D1 中的 32 位数据和 X20 到 X37 中的 24 位数据执行逻辑和运算, 并将结果存储在 D23 和 D24 中。

[梯形图模式]



[列表模式]

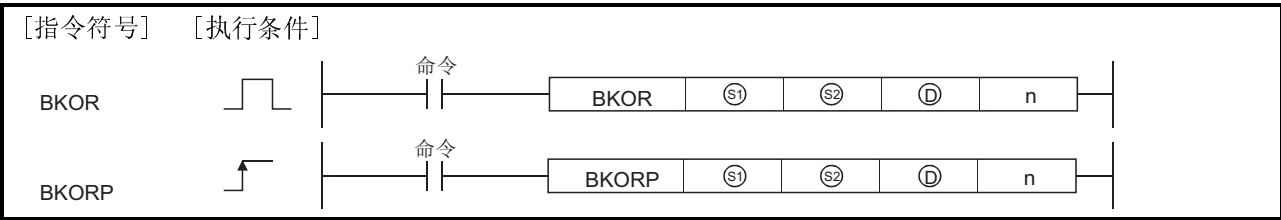
步	指令	软元件
0	LD	M8
1	DORP	D0 K6X20 D23
5	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.1.4 块逻辑和运算(BKOR, BKORP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存 器 Zn
	位	字		位	字		
①	—	○			—		—
②	—	○			—		○
③	—	○			—		—
n	○	○			○		○



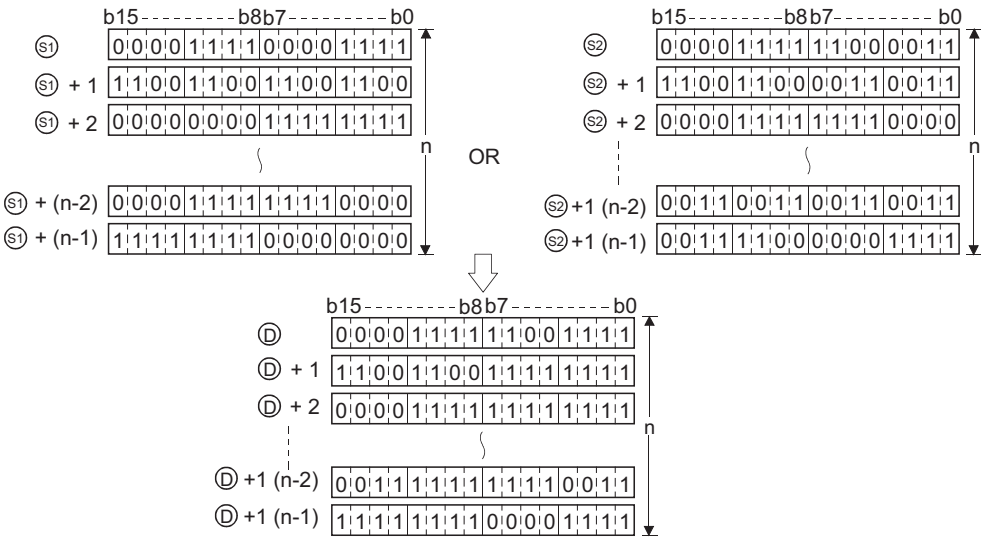
[设定数据]

设定数据	含义	数据类型
①*	存储要用于逻辑和运算的数据的软元件的起始号	BIN 16 位
②*	逻辑运算数据的第一个数或者存储逻辑运算数据的软元件的起始号	
③*	存储运算结果的软元件的起始号	
n	运算数据个数	

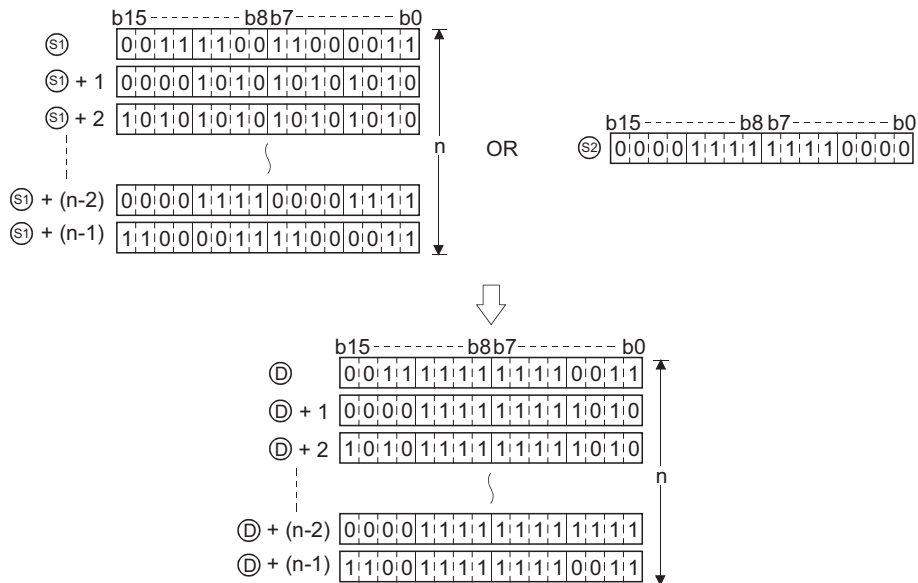
*:可以给①和③或者②和③指定相同的软元件号。

[功能]

(1) 对从①指定的软元件开始的 n 点中的数据 and 从②指定的软元件开始的 n 点中的数据执行逻辑和运算, 并将结果存储在③指定的软元件中。



(2) ②可以指定的常量的范围是-32768 到 32767 (BIN 16 位)。



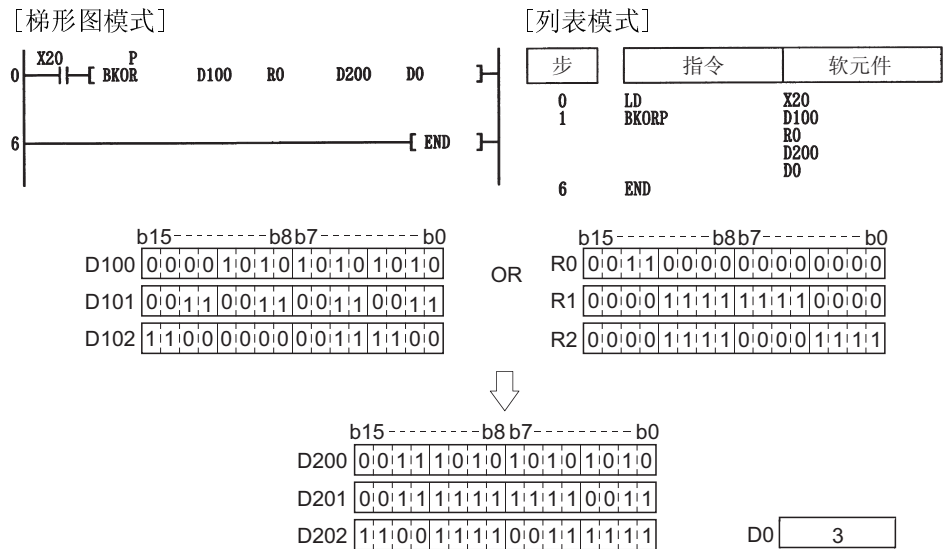
[运行错误]

- (1) 在发生以下运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 从①, ②或④软元件开始的 n 点范围超出了该软元件的范围。 (错误代码: 4101)
 - 从①软元件开始的 n 点软元件范围与从④软元件开始的 n 点软元件范围相互重叠。 (指定给①和④的软元件相同时除外) (错误代码: 4101)
 - 从②软元件开始的 n 点软元件范围与从④软元件开始的 n 点软元件范围相互重叠。 (指定给②和④的软元件相同时除外) (错误代码: 4101)
 - 从①软元件开始的 n 点软元件范围与从②软元件开始的 n 点软元件范围相互重叠。 (错误代码: 4101)

[程序示例]

在本示例中, 程序对下列数据执行逻辑运算:

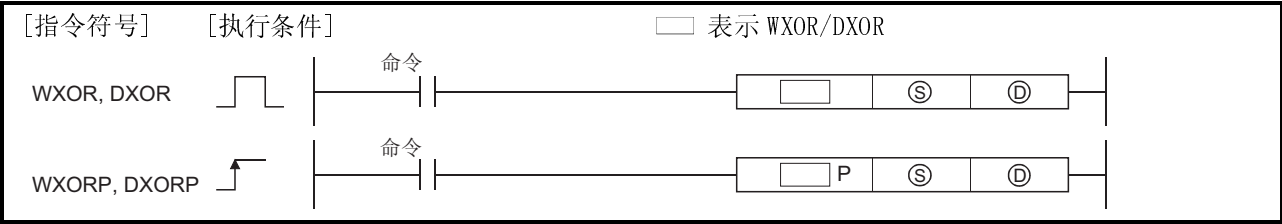
- 从 D100 开始的软元件 (软元件个数由存储在 D0 中的值指定) 中的数据
 - 从 R0 开始的软元件 (软元件个数由存储在 D0 中的值指定) 中的数据
- 然后, 程序将逻辑运算的结果存储在从 D200 开始的软元件 (软元件个数由存储在 D0 中的值指定) 中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.1.5 16 位和 32 位异或运算 (WXOR, WXORP, DXOR, DXORP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	



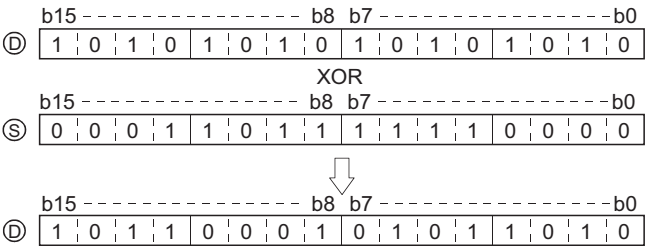
[设定数据]

设定数据	含义	数据类型
⑤	用于执行异或运算的数据, 或者存储这些数据的软元件的起始号	BIN 16/32 位
⑥	存储被执行“异或”运算的数据的软元件的起始号	

[功能]

WXOR

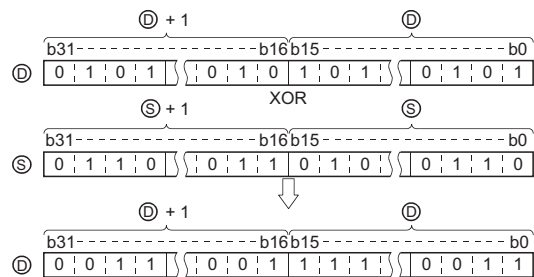
(1) 对⑥指定的软元件中的 16 位数据的每一位与⑤指定的软元件中的 16 位数据的对应位进行异或运算, 并将结果存储在⑥指定的软元件中。



(2) 对位软元件而言, 高于指定位的位数在该运算中被处理为 0。

DXOR

(1) 对⑥指定的软元件中的 32 位数据的每一位与⑤指定的软元件中的 32 位数据的对应位进行异或运算, 并将结果存储在⑥指定的软元件中。



(2) 对位软元件而言, 高于指定位的位数在该运算中被处理为 0。

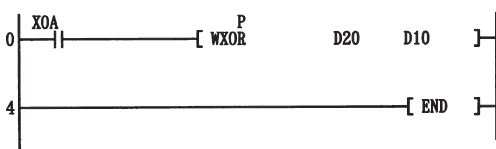
[运行错误]

(1) 如果没有执行变址修改, 则不存在与 WXOR (P) 或 DXOR (P) 指令相关的运行错误。

[程序示例]

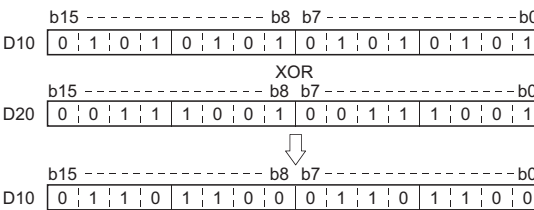
(1) 当 XA 为 ON 时, 以下程序对 D10 和 D20 中的数据执行异或运算, 并将结果存储在 D10 中。

[梯形图模式]



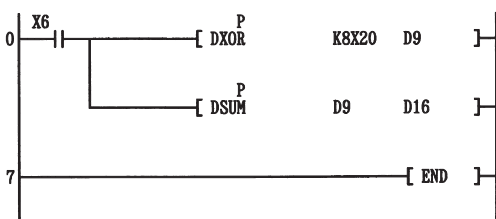
[列表模式]

步	指令	软元件
0	LD	XA
1	WXORP	D20 D10
4	END	



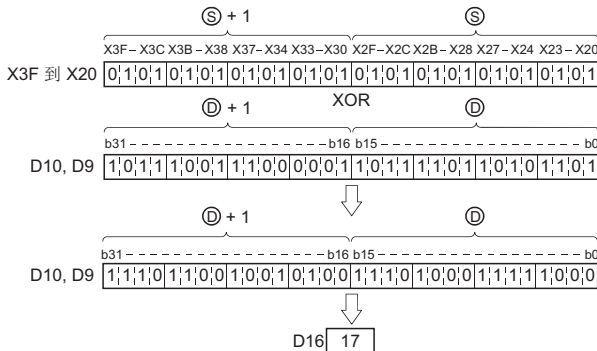
(2) 当 X6 为 ON 时, 以下程序将 X20 到 X3F 中的 32 位数据的位模式与 D9 到 D10 中的数据的数据的位模式进行比较, 并将存在差异的位的数目存储在 D16 中。

[梯形图模式]

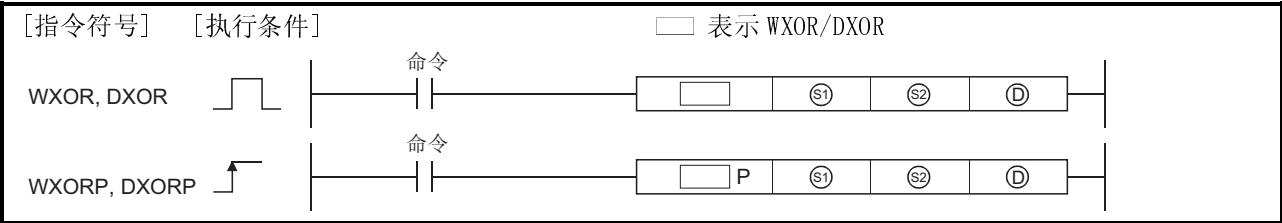


[列表模式]

步	指令	软元件
0	LD	X6
1	DXORP	K8X20 D9
4	DSUMP	D9 D16
7	END	



设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
①	○						○	—	
②	○						○	—	
③	○						—	—	



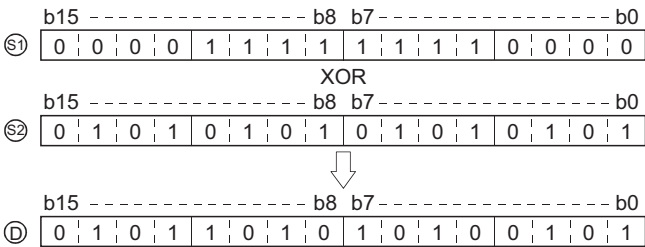
[设定数据]

设定数据	含义	数据类型
①	用于执行异或运算的数据, 或者存储这些数据的软元件的起始号	BIN 16/32 位
②		
Ⓓ	存储异或运算结果的软元件的起始号	

[功能]

WXOR

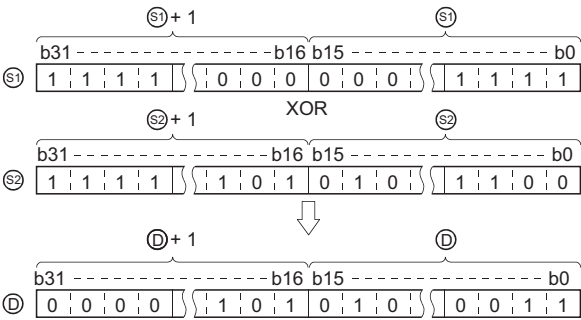
(1) 对①指定的软元件中的 16 位数据的每一位与②指定的软元件中的 16 位数据的对应位进行异或运算, 并将结果存储在Ⓓ指定的软元件中。



(2) 对位软元件而言, 高于指定位的位数在该运算中被处理为 0。[参见程序示例(2)]

DXOR

(1) 对①指定的软元件中的 32 位数据的每一位与②指定的软元件中的 32 位数据的对应位进行异或运算, 并将结果存储在Ⓓ指定的软元件中。



(2) 对位软元件而言, 高于指定位的位数在该运算中被处理为 0。

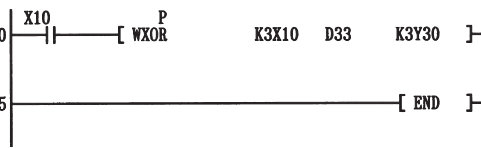
[运行错误]

(1) 不存在与 WXOR (P) 或 DXOR (P) 指令相关的运行错误。

[程序示例]

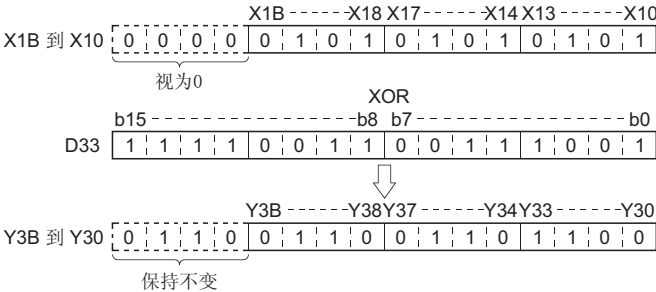
(1) 当 X10 为 ON 时, 以下程序对 X10 到 X1B 中的数据和 D33 中的数据执行异或运算, 并将结果输出到 Y30 到 Y3B 中。

[梯形图模式]



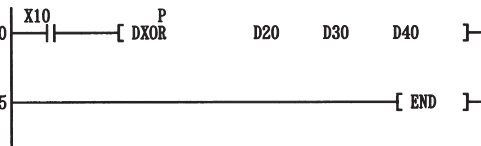
[列表模式]

步	指令	软元件
0	LD	X10
1	WXORP	K3X10 D33 K3Y30
5	END	



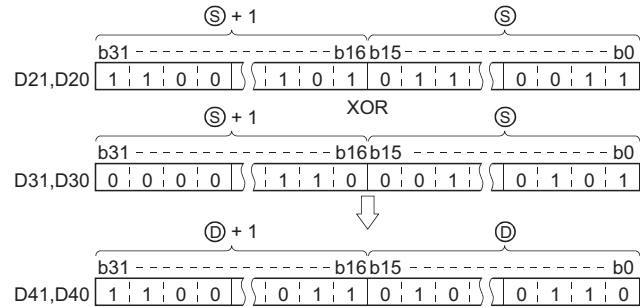
(2) 当 X10 为 ON 时, 以下程序对 D20, D21 中的数据和 D30, D31 中的数据执行异或运算, 并将结果存储在 D40 和 D41 中。

[梯形图模式]



[列表模式]

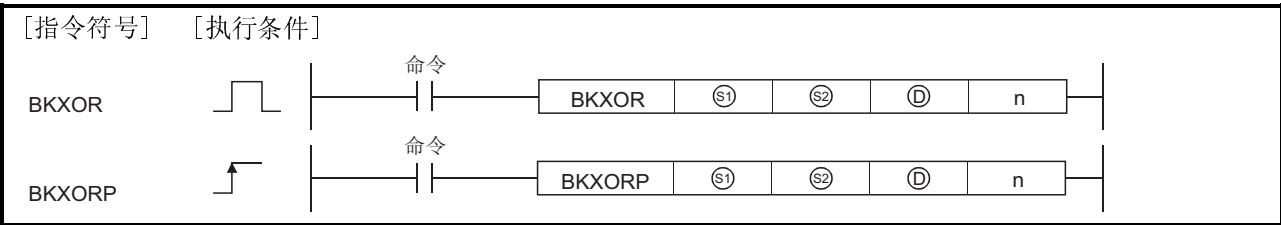
步	指令	软元件
0	LD	X10
1	DXORP	D20 D30 D40
5	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.1.6 块异或运算 (BKXOR, BKXORP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [X][Y]		特殊功能模 块 [I][O][G]	变址寄存器 Zn
	位	字		位	字		
①	—	○			—		—
②	—	○			—		○
③	—	○			—		—
n	○	○			○		○



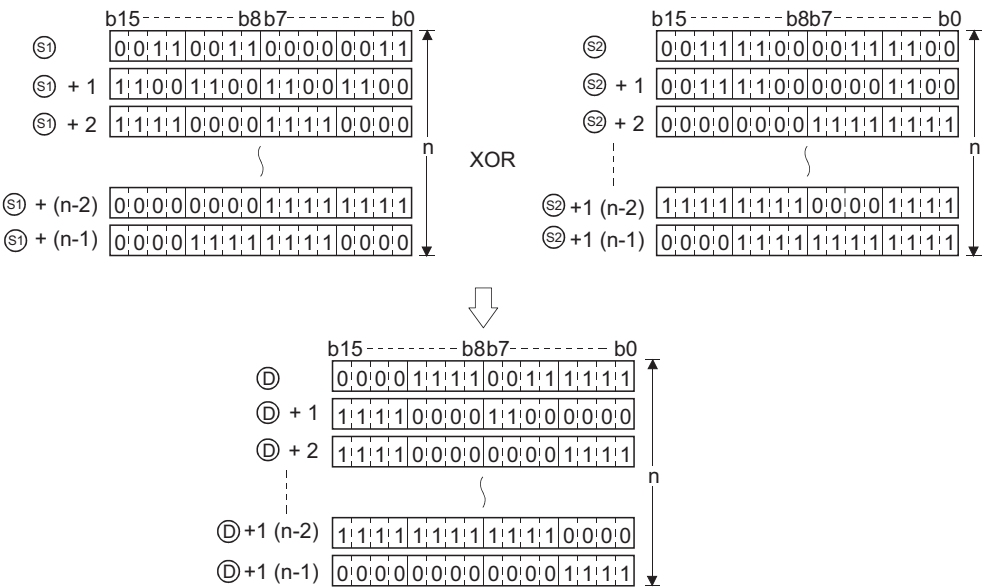
[设定数据]

设定数据	含义	数据类型
①*	存储要用于异或运算的数据的软元件的起始号	BIN 16 位
②*	逻辑运算数据的第一个数或者存储逻辑运算数据的软元件的起始号	
③*	存储运算结果的软元件的起始号	
n	存储要用于异或运算的数据的个数	

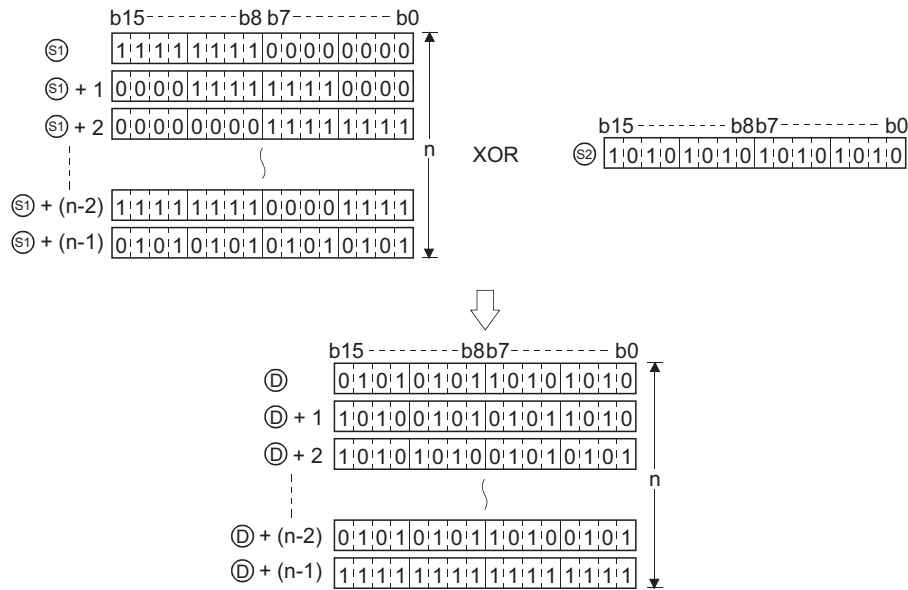
*: 可以给①和③或者②和③指定相同的软元件号。

[功能]

(1) 对从①指定的软元件开始的 n 点中的数据 and 从②指定的软元件开始的 n 点中的数据执行异或运算, 并将结果存储在③指定的软元件中。



(2) S2可以指定的常量的范围是-32768 到 32767 (BIN 16 位)

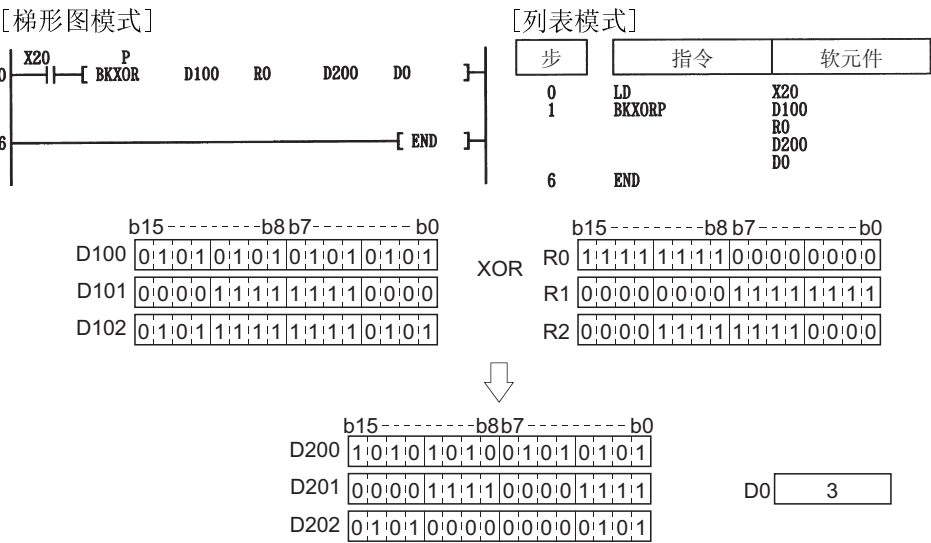


[运行错误]

- (1) 在发生以下运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 从 S1, S2或D软元件开始的 n 点范围超出了该软元件的范围。 (错误代码: 4101)
 - 从 S1软元件开始的 n 点软元件范围与从 D软元件开始的 n 点软元件范围相互重叠。 (指定给 S1和D) 的软元件相同时除外) (错误代码: 4101)
 - 从 S2软元件开始的 n 点软元件范围与从 D软元件开始的 n 点软元件范围相互重叠。 (指定给 S2和D) 的软元件相同时除外) (错误代码: 4101)
 - 从 S2软元件开始的 n 点软元件范围与从 S2软元件开始的 n 点软元件范围相互重叠。 (错误代码: 4101)

[程序示例]

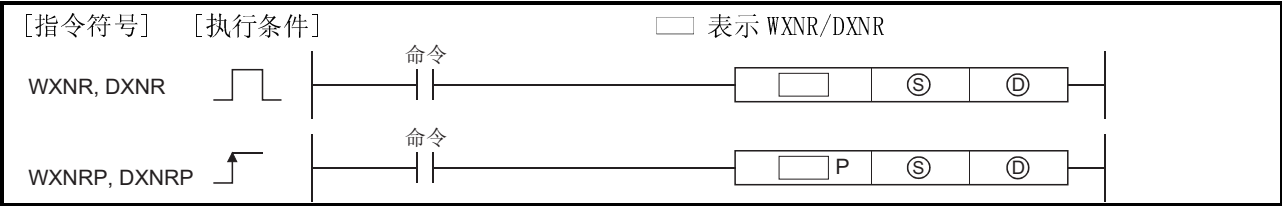
- (1) 在本示例中, 程序对下列数据执行异或运算:
- 从 D100 开始的软元件 (软元件个数由存储在 D0 中的值指定) 中的数据
 - 从 R0 开始的软元件 (软元件个数由存储在 D0 中的值指定) 中的数据
- 然后, 程序将异或运算的结果存储在从 D200 开始的软元件 (软元件个数由存储在 D0 中的值指定) 中。



QCPU			QnA	Q4AR
PLC CPU		处理CPU		
基本	高性能			
○	○	○	○	○

7.1.7 16 位和 32 位异或非运算(WXNR, WXNRP, DXNR, DXNRP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]G[][]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	



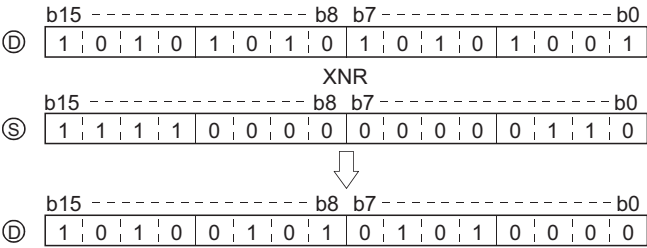
[设定数据]

设定数据	含义	数据类型
Ⓢ	将用于执行异或非运算的数据, 或者存储这些数据的软元件的起始号	BIN 16/32 位
Ⓓ	存储被执行“异或非”运算的数据的软元件的起始号	

[功能]

WXNR

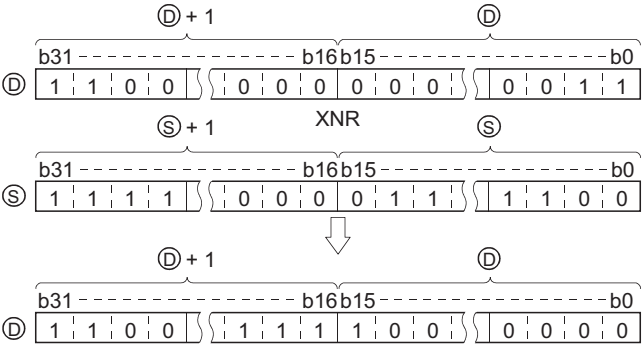
- (1) 对Ⓓ指定的软元件中的 16 位数据的每一位与Ⓢ指定的软元件中的 16 位数据的对应位进行异或非运算, 并将结果存储在Ⓓ指定的软元件中。



- (2) 对位软元件而言, 高于指定位的位数在该运算中被处理为 0。

DXNR

(1) 对Ⓓ指定的软元件中的 32 位数据的每一位与Ⓔ指定的软元件中的 32 位数据的对应位进行异或非运算, 并将结果存储在Ⓓ指定的软元件中。



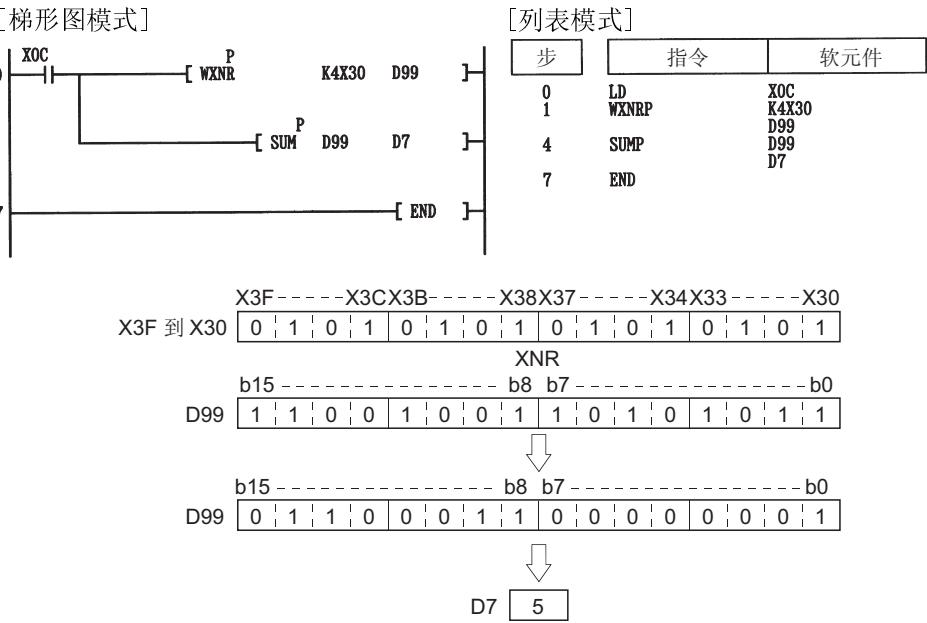
(2) 对位软元件而言, 高于指定位的位数在该运算中被处理为 0。

[运行错误]

(1) 不存在与 WXNR (P) 或 DXNR (P) 指令相关的运行错误。

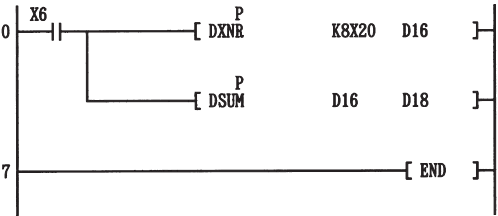
[程序示例]

(1) 当 Xc 变为 ON 时, 以下程序将 X30 到 X3F 中的 16 位数据的位模式与 D99 中的数据位模式进行比较, 并将相同模式的数目存储在 D7 中。



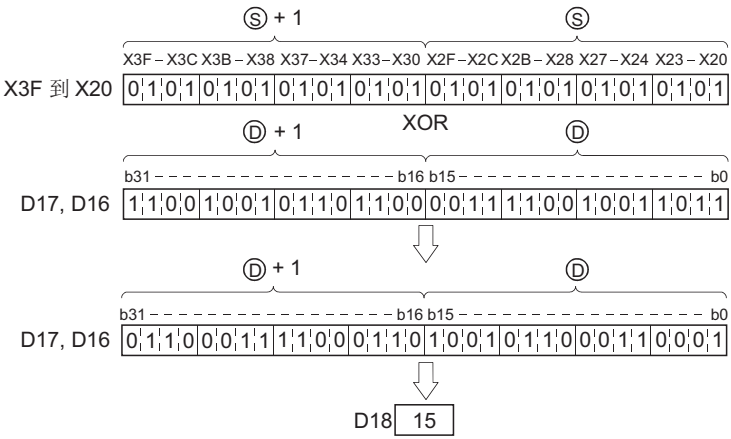
(2) 当 X6 为 ON 时, 以下程序将 X20 到 X3F 中的 32 位数据的位模式与 D16, D17 中的数据位模式进行比较, 并将相同位模式的数目存储在 D18 中。

[梯形图模式]



[列表模式]

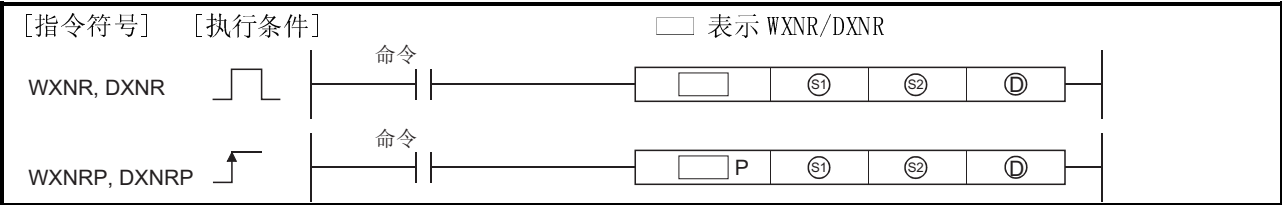
步	指令	软元件
0	LD	X6
1	DXNRP	K8X20 D16
4	DSUMP	D16 D18
7	END	



注释

*:有关 SUMP/DSUM 指令的更多信息, 请参阅 7.5.2 节。

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[K]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
Ⓔ1	○							○	—
Ⓔ2	○							○	—
Ⓓ	○							—	—



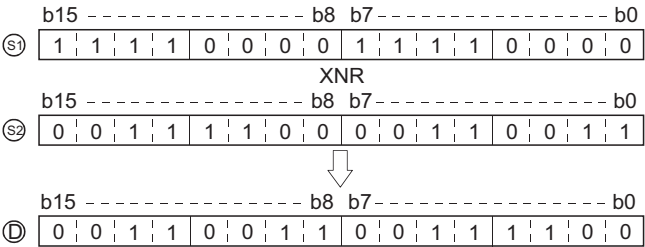
[设定数据]

设定数据	含义	数据类型
Ⓢ1	用于执行异或非运算的数据, 或者存储这些数据的软元件的起始号	BIN 16/32 位
Ⓢ2		
Ⓓ	将用于存储异或非运算结果的软元件的起始号	

[功能]

WXNR

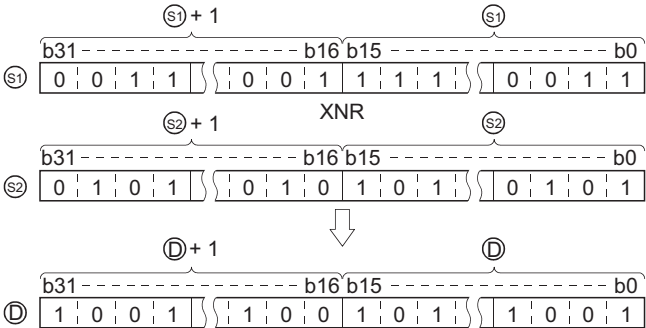
(1) 对Ⓢ1指定的软元件中的 16 位数据的每一位与Ⓢ2指定的软元件中的 16 位数据的对应位进行异或非运算, 并将结果存储在Ⓓ指定的软元件中。



(2) 对位软元件而言, 高于指定位的位数在该运算中被处理为 0。

DXNR

(1) 对①指定的软元件中的 32 位数据的每一位与②指定的软元件中的 32 位数据的对应位进行异或非运算, 并将结果存储在③指定的软元件中。



(2) 对位软元件而言, 高于指定位的位数在该运算中被处理为 0。

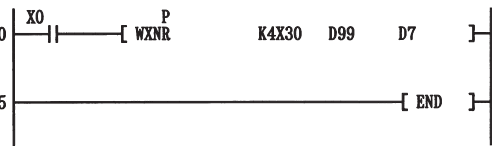
[运行错误]

(1) 不存在与 WXNR (P) 或 DXNR (P) 指令相关的运行错误。

[程序示例]

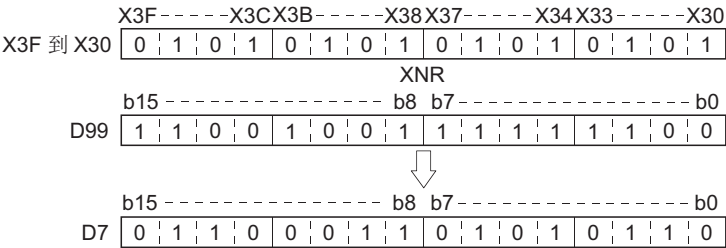
(1) 当 X0 为 ON 时, 以下程序对 X30 到 X3F 中的 16 位数据和 D99 中的数据执行异或非运算, 并将结果存储在 D7 中。

[梯形图模式]

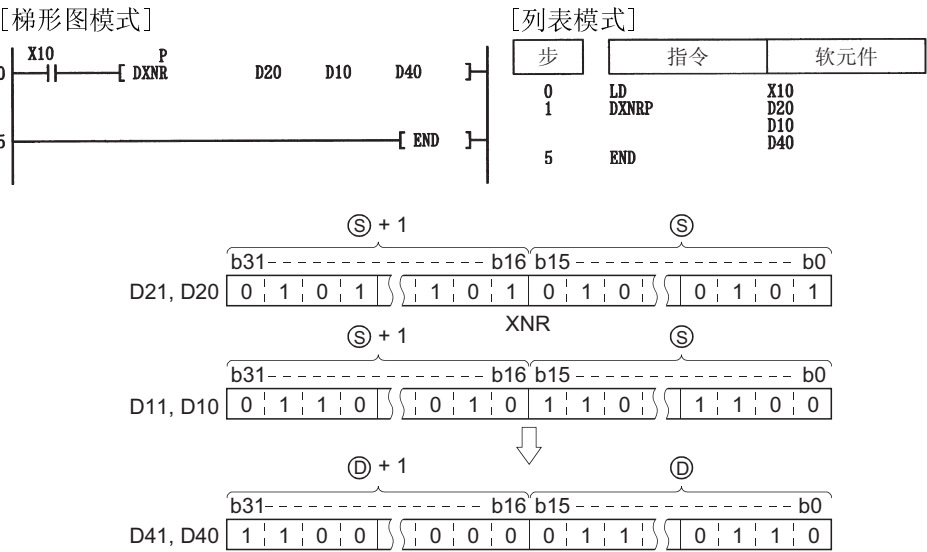


[列表模式]

步	指令	软元件
0	LD	X0
1	WXNRP	K4X30 D99 D7
5	END	



(2) 当 X10 为 ON 时, 以下程序对 D20, D21 中的 32 位数据和 D10, D11 中的数据执行异或非运算, 并将结果存储在 D40 和 D41 中。



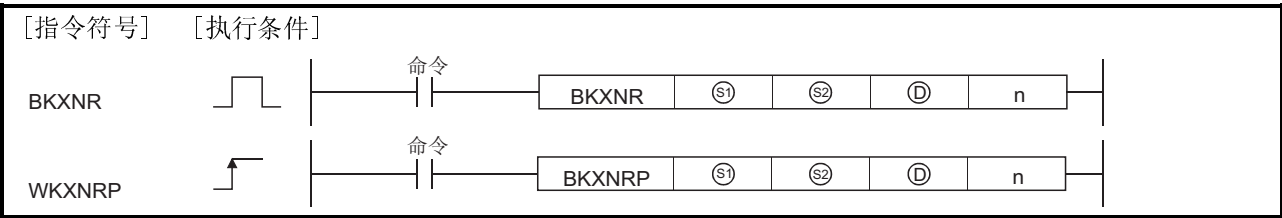
注释

*: 有关 SUM/DSUMP 指令的更多信息, 请参阅 7.5.2 节。

QCPU			QnA	Q4AR
PLC CPU		过程CPU		
基本型	高性能型			
○	○	○	○	○

7.1.8 块异或非运算(BKXNR, BKXNRP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
①	—	○			—		—
②	—	○			—		○
③	—	○			—		—
n	○	○			○		○



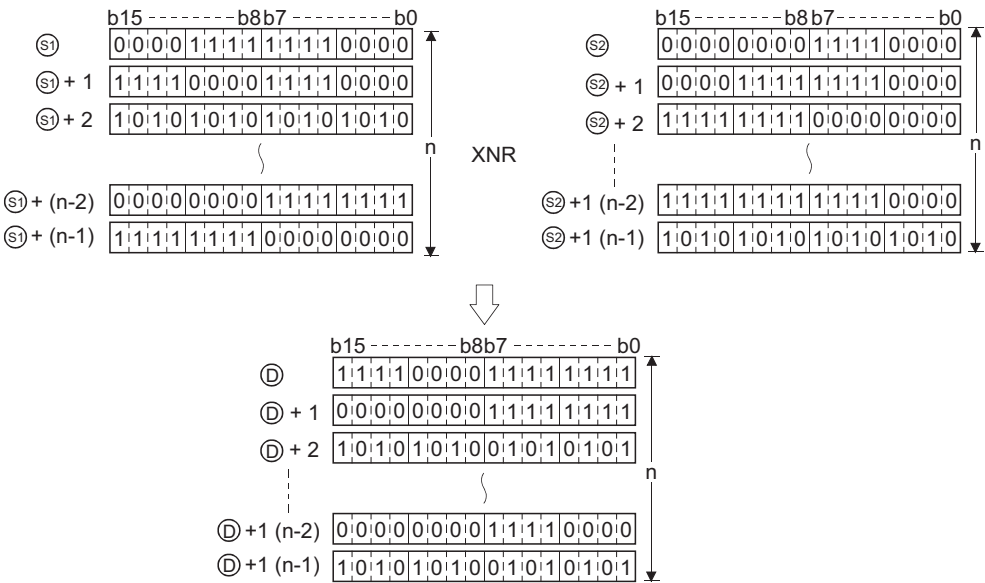
[设定数据]

设定数据	含义	数据类型
①*	存储要用于异或非运算的数据的软元件的起始号	BIN 16 位
②*	逻辑运算数据的第一个数或者存储逻辑运算数据的软元件的起始号	
③*	存储运算结果的软元件的起始号	
n	运算数据个数	

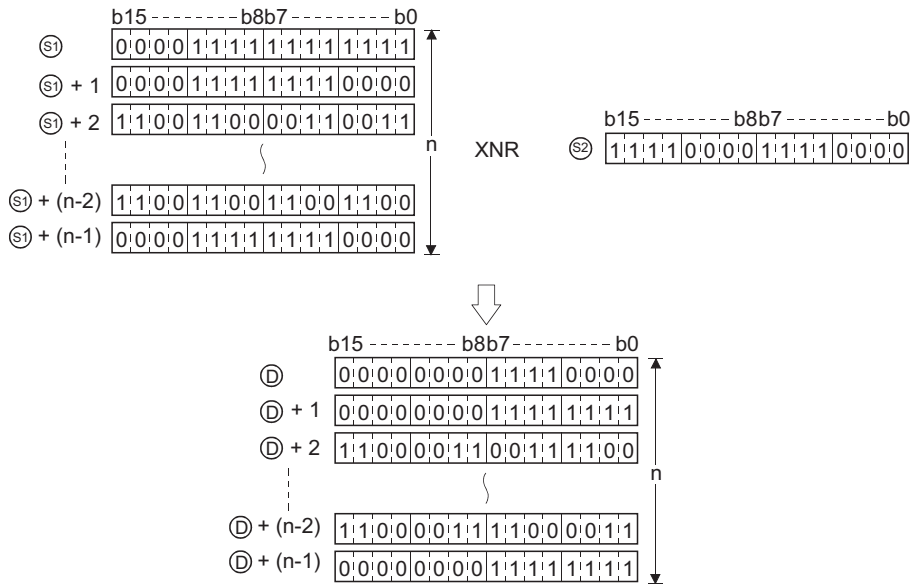
*: 可以给①和③或者②和③指定相同的软元件号。

[功能]

(1) 对从①指定的软元件开始的 n 点中的数据和从②指定的软元件开始的 n 点中的数据执行异或非运算, 并将结果存储在③指定的软元件中。



(2) ⑤2可以指定的常量的范围是-32768 到 32767 (BIN 数据)



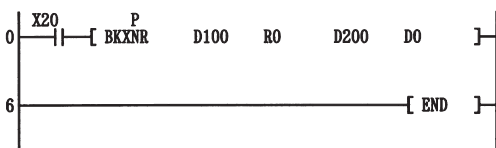
[运行错误]

- (1) 在发生以下运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 从 ⑤1, ⑤2或 ⑤软元件开始的 n 点范围超出了该软元件的范围。 (错误代码: 4101)
 - 从 ⑤1软元件开始的 n 点软元件范围与从 ⑤软元件开始的 n 点软元件范围相互重叠。 (指定给 ⑤1和 ⑤的软元件相同时除外) (错误代码: 4101)
 - 从 ⑤2软元件开始的 n 点软元件范围与从 ⑤软元件开始的 n 点软元件范围相互重叠。 (指定给 ⑤2和 ⑤的软元件相同时除外) (错误代码: 4101)
 - 从 ⑤1软元件开始的 n 点软元件范围与从 ⑤2软元件开始的 n 点软元件范围相互重叠。 (错误代码: 4101)

[程序示例]

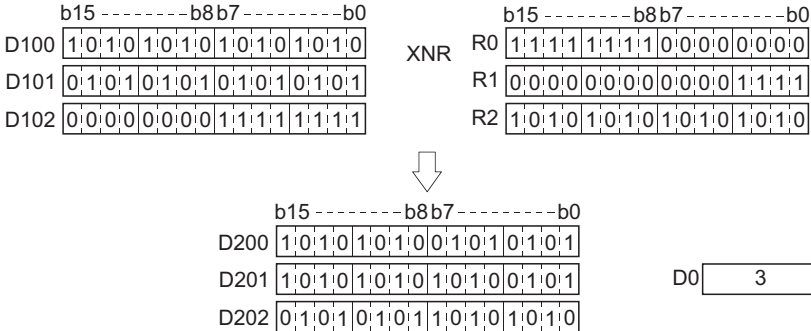
- (1) 在本示例中, 程序对下列数据执行异或非运算:
- 从 D100 开始的软元件 (软元件个数由存储在 D0 中的值指定) 中的数据。
 - 从 R0 开始的软元件 (软元件个数由存储在 D0 中的值指定) 中的数据, 然后, 程序将异或非运算的结果存储在从 D200 开始的软元件 (软元件个数由存储在 D0 中的值指定)。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X20
1	BKXNRP	D100 R0 D200 D0
6	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.2 循环指令

7.2.1 16 位数据的右循环(ROR, RORP, RCR, RCRP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
①	○						—	—	
n	○						○	—	

[指令符号]	[执行条件]	□ 表示 ROR/RCR
ROR, RCR		
RORP, RCRP		

[设定数据]

设定数据	含义	数据类型
①	执行循环的软元件的起始号	BIN 16 位
n	循环位数(0 到 15)	

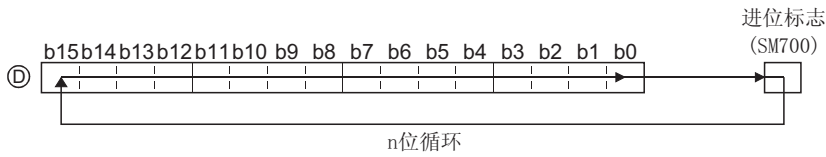
[功能]

ROR

- (1) 将由①指定的软元件的 16 位数据，不包括进位标志，往右循环移动 n 位。
ROR 指令执行之前的状态决定进位标志是 ON 或 OFF。
-
- (2) 当位软元件被在①处指定后，数据循环移动指定的位数。
在这种情况下，实际循环的位数是 n 除以指定位数所得商的余数。
例如，当 n=15 且指定的位数是 12 位，则 15 除以 12 的商为 1，余数为 3，因此将循环移动 3 位。
- (3) 将 0 到 15 之间的任意一值指定为给 n。
当指定的 n 值为 16 或更大时，则 n 除以 16 的余数用于循环。
例如，当 n=18，18 除以 16 的余数为 2，则向右循环移动 2 位。

RCR

- (1) 将由①指定的软元件的 16 位数据，包括进位标志，向右循环移动 n 位。
进位标志依据于它在 ROR 指令执行前的状态，变为 ON 或 OFF。



- (2) 当一个位软元件被在①处指定时，数据将循环移动指定的位数。
在这种情况下，实际循环的位数是 n 除以指定位数所得商的余数。
例如，当 $n=15$ 并且指定的位数是 12 位，则 15 除以 12 的商为 1，余数为 3，因此将循环移动 3 位。
- (3) 将 0 到 15 之间的任意一值指定给 n 。
当指定的 n 值为 16 或更大，则 n 除以 16 的余数用于循环。
例如，当 $n=18$ ，18 除以 16 的余数为 2，则向右循环移动 2 位。

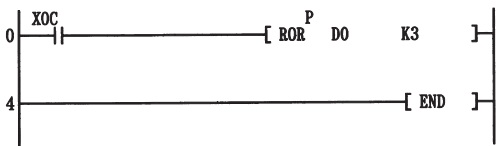
[运行错误]

- (1) 没有与 ROR(P) 或 RCR(P) 指令有关的运行错误。

[程序示例]

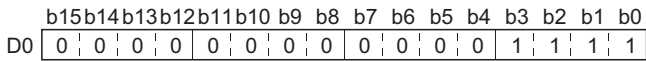
- (1) 当 X_c 为 ON 时，下列程序将 D0 的内容，不包括进位标志，向右循环移动 3 位。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	XOC
1	RORP	D0 K3
4	END	



进位标志 (SM700)

0

进位标志 (SM700)

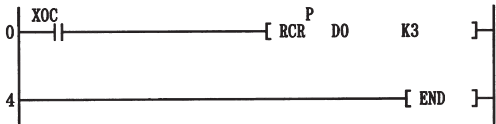
1

执行前b0到b2的内容 执行前b4到b15的内容 执行前b3的内容

执行前b2的内容

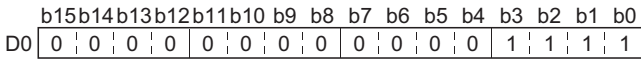
- (2) 当 X_c 为 ON 时，下列程序将 D0 的内容，包括进位标志，向右循环移动 3 位。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	XOC
1	RCRP	D0 K3
4	END	



进位标志 (SM700)

*

进位标志 (SM700)

1

执行前b1和b0的内容 进位标志SM700内容 执行前b4到b15的内容 执行前b3内容 执行前b2内容

* 进位标志依据它在RCR执行前的状态，变为ON/OFF。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.2.2 16 位数据左循环 (ROL, ROLP, RCL, RCLP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\A[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
①	○						—	—	
n	○						○	—	

[指令符号]	[执行条件]	□ 表示 ROL/RCL	
ROL, RCL			
ROLP, RCLP			

[设定数据]

设定数据	含义	数据类型
①	执行循环的软元件的起始号	BIN 16 位
n	循环位数 (0 到 15)	

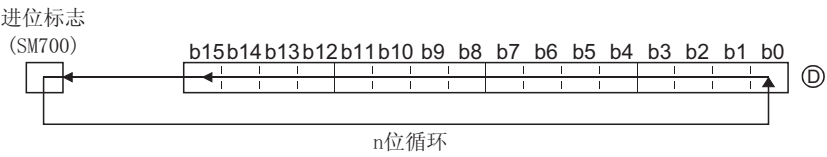
[功能]

ROL

- (1) 将①指定的软元件的 16 位数据，不包括进位标志，向左循环移动 n 位。
ROL 指令执行前的状态决定进位标志是 ON 或 OFF。
- 进位标志
(SM700)
- (2) 当一个位软元件在①处被指定时，数据循环移动指定的位数。
在这种情况下，实际循环的位数是 n 除以指定位数所得的余数。
例如，当 n=15 并且指定的位数是 12 位，则 15 除以 12 的商为 1，余数为 3，因此将循环移动 3 位。
- (3) 将 0 到 15 中的任意一个指定给 n。
如果指定给 n 的值为 16 或更大，则 n 除以 16 的余数用于循环。
例如，当 n=18 时，因为余数为 2，因此将向右循环移动 2 位。

RCL

- (1) 将①指定的软元件的 16 位数据，包括进位标志，向左循环移动 n 位。
进位标志依据于它在 RCL 指令执行前的状态，变为 ON 或 OFF。



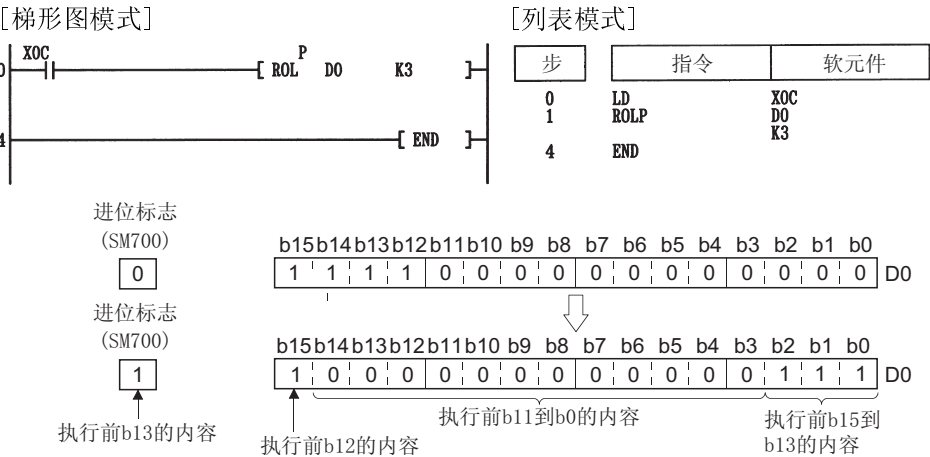
- (2) 当一个位软元件在⑤处被指定时，对数据循环移动指定的位数。
在这种情况下，实际循环移动的位数是 n 除以指定位数所得的余数。
例如，当 $n=15$ 且指定位数为 12 位，则 15 除以 12 的商为 1，余数为 3，因此 3 位循环。
- (3) 将 0 到 15 中的任意一位指定给 n 。
如果指定的 n 的值为 16 或更大，则 n 除以 16 的余数用于循环。
例如，当 $n=18$ 时，因为余数为 2，因此将向右循环移动 2 位。

[运行错误]

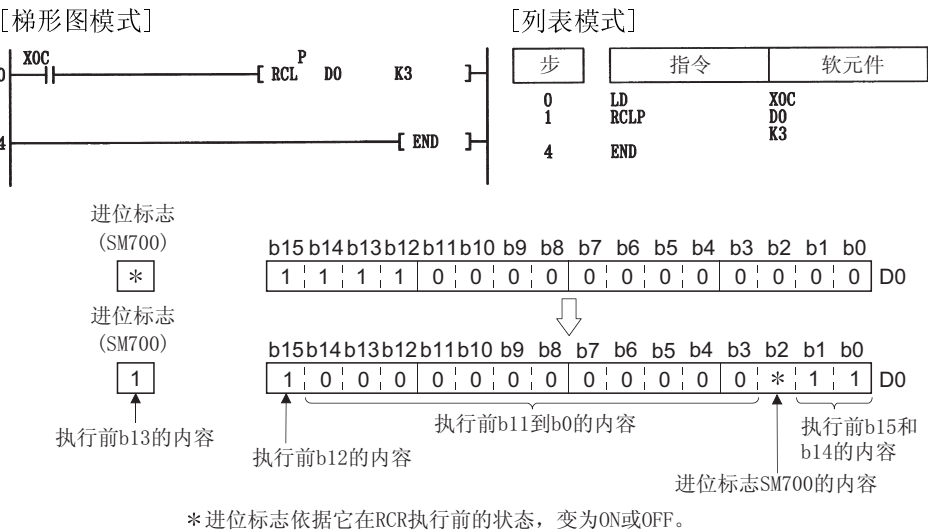
- (1) 没有与 ROL(P) 或 RCL(P) 指令有关的运行错误。

[程序示例]

- (1) 当 Xc 为 ON 时，下列程序将 D0 的内容，不包括进位标志，向左循环移动 3 位。



- (2) 当 Xc 为 ON 时，下列程序将 D0 的内容，包括进位标志，向左循环移动 3 位。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.2.3 32 位数据的右循环 (DROR, DRORP, DRCR, DRCRP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]N[]		特殊功能模 块 U[]G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
①	○						—	—	
n	○						○	—	

[指令符号]	[执行条件]	□ 表示 DROR/DRCR
DROR, DRCR		
DRORP, DRCRP		

[设定数据]

设定数据	含义	数据类型
①	执行循环的软元件的第一个软元件号	BIN 32 位
n	循环位数 (0 到 31)	BIN 16 位

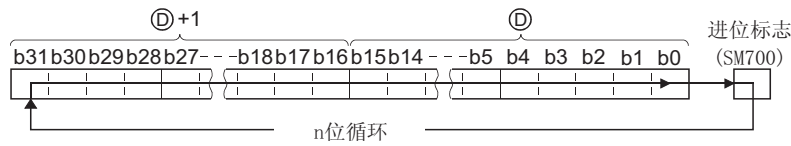
[功能]

DROR

- (1) 将①指定的软元件的 32 位数据，不包括进位标志，向右侧循环移动 n 位。
DROR 指令执行前的状态决定进位标志为 ON 或 OFF。
-
- (2) 当一个位软元件在①处被指定时，数据将循环移动指定的位数。
在这种情况下，实际循环的位数是 n 除以指定位数所得商的余数。
例如，如果 n=31 且数字指定位所指定的位数为 24 位，则有 7 位循环，因为计算公式为 31 除以 24 等于 1，余数为 7。
- (3) 将 0 到 31 中的任意一位指定给 n。
如果指定的 n 值是 32 或更大，则 n 除以 32 的余数用于循环。
例如，当 n=34 时，因为 34 除以 32 的余数为 2，因此向右循环移动 2 位。

DRCR

- (1) 在①指定的软元件中的 32 位数据，包括进位标志，向右循环移动 n 位。
进位标志依据于它在 DRCR 指令执行前的状态，变为 ON 或 OFF。



- (2) 当一个位软元件在Ⓓ被指定时，数据循环移动指定位数。
在这种情况下，实际循环的位数是 n 除以指定位数所得商的余数。
例如，如果 n=31 并且指定位所指定的位数是 24 位，则将进行 7 位循环，因为计算公式是 31 除以 24 所得商为 1，余数是 7。
- (3) 将 0 到 31 中的任意一个指定给 n。
如果指定的 n 值是 32 或更大，则 n 除以 32 的余数用于循环。
例如，当 n=34 时，因为 34 除以 32 的余数为 2，因此 2 位循环到右侧。

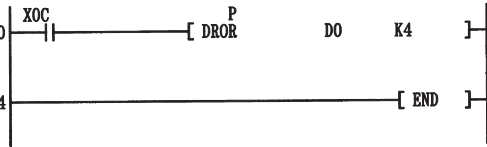
[运行错误]

- (1) 没有与 DROR (P) 或 DRCR (P) 指令相关的运行错误。

[程序示例]

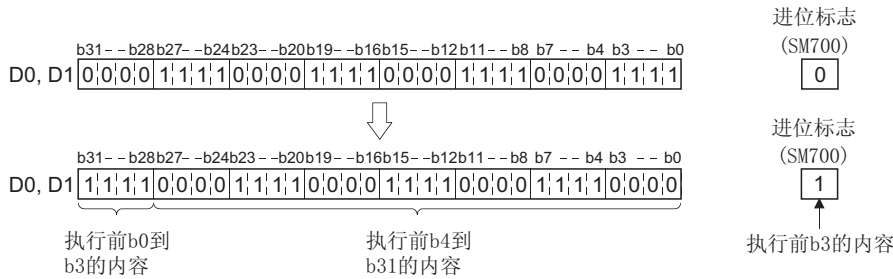
- (1) 当 Xc 为 ON 时，下列程序将 D0 和 D1 的内容，不包括进位标志，向右循环移动 4 位。

[梯形图模式]



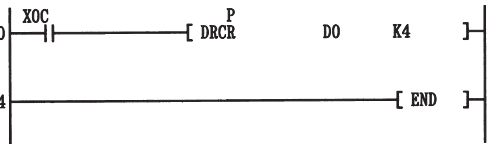
[列表模式]

步	指令	软元件
0	LD	XOC
1	DRORP	D0 K4
4	END	



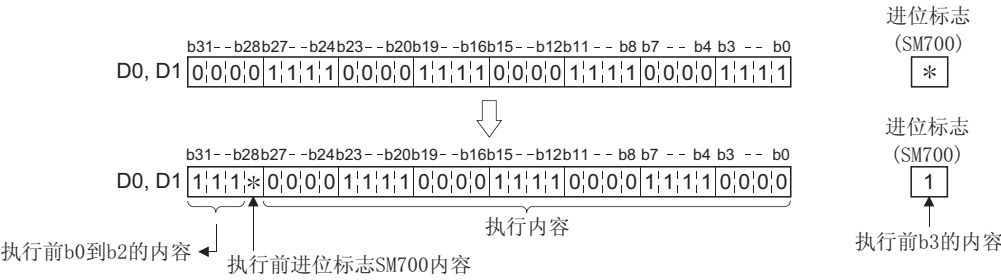
- (2) 当 Xc 为 ON 时，下列程序将 D0 和 D1 的内容，包括进位标志，向右循环移动 4 位。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	XOC
1	DRCRP	D0 K4
4	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.2.4 32 位数据左循环 (DROL, DROL, DRCL, DRCLP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		常数 E
①							—
n							○

[指令符号]	[执行条件]	□ 表示 DROL/DRCL
DROL, DRCL		
DROLP, DRCLP		

[设定数据]

设定数据	含义	数据类型
①	执行循环的软元件的第一个软元件号	BIN 32 位
n	循环位数 (0 到 31)	BIN 16 位

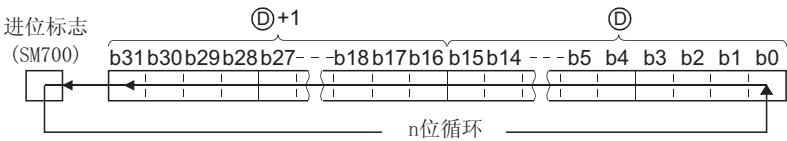
[功能]

DROL

- (1) 将①指定的软元件的 32 位数据，不包括进位标志，向左循环移动 n 位。
进位标志依据于它在执行 DROL 指令之前的状态，变为 ON 或 OFF。
-
- (2) 当一个位软元件在①处被指定时，数据循环移动指定的位数。
在这种情况下，实际循环的位数是 n 除以指定位数所得商的余数。
例如，如果 n=31 并且指定位所指定的位数是 24 位，则将循环移动 7 位，因为计算公式是 31 除以 24 所得商为 1，余数是 7。
- (3) 将 0 到 31 中的任意一个指定为 n。
如果指定的 n 值是 32 或更大，则 n 除以 32 的余数用于循环。
例如，当 n=34 时，因为 34 除以 32 的余数为 2，因此 2 位循环到右侧。

DRCL

- (1) 在①指定的软元件中，将 32 位数据，包括进位标志，向左循环移动 n 位。
进位标志依据于它在 DRCL 指令执行前的状态，变为 ON 或 OFF。



- (2) 当一个位软元件在Ⓓ处被指定时，数据循环移动指定的位数。
在这种情况下，实际进行循环的位数是 n 除以指定位数所得商的余数。
例如，如果 $n=31$ 并且位指定所指定的位数是 24 位，则将循环移动 7 位，因为计算公式是 31 除以 24 所得商为 1，余数是 7。
- (3) 将 0 到 31 中的任意一个指定给 n 。
如果指定的 n 值是 32 或更大，则 n 除以 32 的余数用于循环。
例如，当 $n=34$ 时，因为 34 除以 32 的余数为 2，因此 2 位循环到右侧。

[运行错误]

- (1) 没有与 DROL(P) 或 DRCL(P) 指令相关的运行错误。

[程序示例]

- (1) 当 Xc 为 ON 时，下列程序将 D0 和 D1 的内容，不包括进位标志，向左循环移动 4 位。

[梯形图模式]

[列表模式]

步	指令	软元件
0	LD	XOC
1	DROLP	D0 K4
4	END	

进位标志 (SM700) 0

进位标志 (SM700) 1

执行前b28的内容

执行前b27到b0的内容

执行前b31到b28的内容

b31--b28b27--b24b23--b20b19--b16b15--b12b11--b8b7--b4b3--b0

1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 D0, D1

0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 D0, D1

- (2) 当 Xc 为 ON 时，下列程序循环将 D0 和 D1 的内容，包括进位标志，向左循环移动 4 位。

[梯形图模式]

[列表模式]

步	指令	软元件
0	LD	XOC
1	DRCLP	D0 K4
4	END	

进位标志 (SM700) *

进位标志 (SM700) 1

执行前b28的内容

执行前b27到b0的内容

执行前b31到b29的内容

执行前进位标志SM700的内容

b31--b28b27--b24b23--b20b19--b16b15--b12b11--b8b7--b4b3--b0

1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 D0, D1

0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 * 1 1 1 1 D0, D1

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.3 移位指令

7.3.1 16 位数据的 n 位左移或右移 (SFR, SFRP, SFL, SFLP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[] []		特殊功能模 块 U[] G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
⑤	○						—	—	
n	○						○	—	

[指令符号]	[执行条件]	□ 表示 SFR/SFL	
SFR, SFL			
SFRP, SFLP			

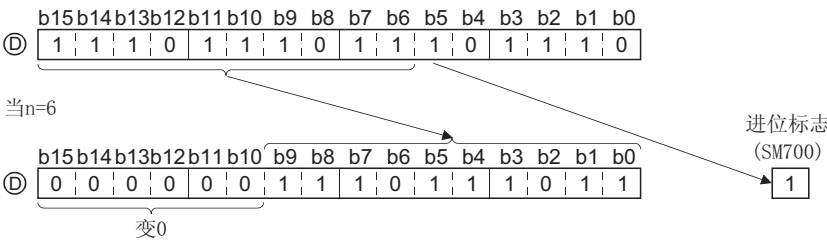
[设定数据]

设定数据	含义	数据类型
Ⓓ	存储要移位数据的软元件的第一个软元件号	BIN 16 位
n	移位位数 (0 到 15)	

[功能]

SFR

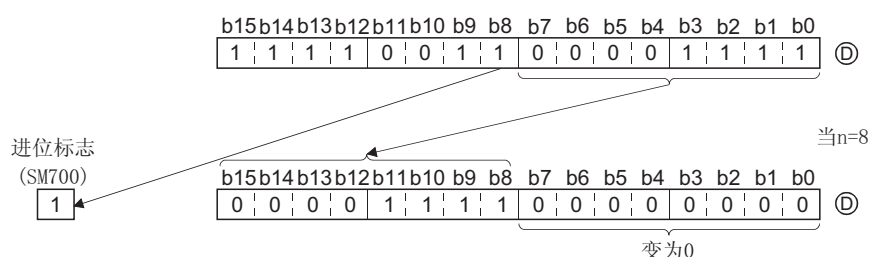
(1) 将Ⓓ指定的软元件中的 16 位数据，右移 n 位。



- (2) 从最高位开始的总共 n 位变为 0。
- (3) 当一个字软元件在Ⓓ处被指定时，数据移位指定位数。[见程序示例(1)]
- (4) 将 0 到 15 中的任意一个指定给 n。
如果指定的 n 值是 16 或更大，则 n 除以 16 的余数用于循环。
例如，当 n=18，则有向右移动 2 位，因为 18 除以 16 的余数为 2。

SFL

(1) 将①指定的软元件中的 16 位数据，左移 n 位。



(2) 从最低位开始的 n 位变为 0。

(3) 当一个字软元件在②处被指定时，数据移位指定的位数。[见程序示例(2)]

(4) 将 0 到 15 中的任意一个指定给 n。

如果指定的 n 值是 16 或更大，则 n 除以 16 的余数用于循环。

例如，当 n=18，则有 2 位循环到右侧，因为 18 除以 16 的余数为 2。

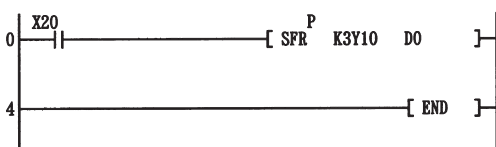
[运行错误]

(1) 没有与 SFR 或 SFL(P) 指令相关的运行错误。

[程序示例]

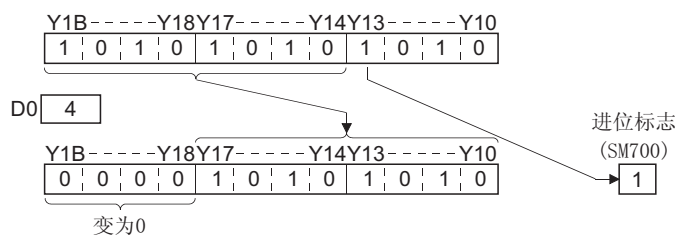
(1) 在 X20 为 ON 时，下列程序将 Y10 到 Y1B 的内容往右移位 D0 指定的位数。

[梯形图模式]



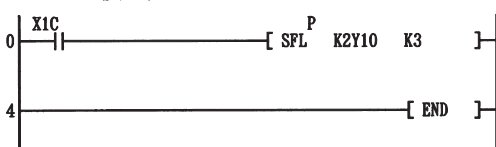
[列表模式]

步	指令	软元件
0	LD	X20
1	SFRP	K3Y10 D0
4	END	



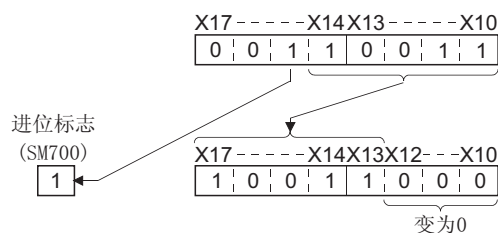
(2) 当 X1C 为 ON 时，下列程序将 X10 到 X17 内容的向左移位 3 位。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X1C
1	SFLP	K2Y10 K3
4	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.3.2 n 位数据的 1 位左移或右移 (BSFR, BSFRP, BSFL, BSFLP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [J][N][C]		特殊功能模 块 [U][G][C]	变址寄存器 Zn
	位	字		位	字		常数 E
①	○	—					—
n	—	○					—

[指令符号]	[执行条件]	□ 表示 BSFR/BSFL
BSFR, BSFL		
BSFRP, BSFLP		

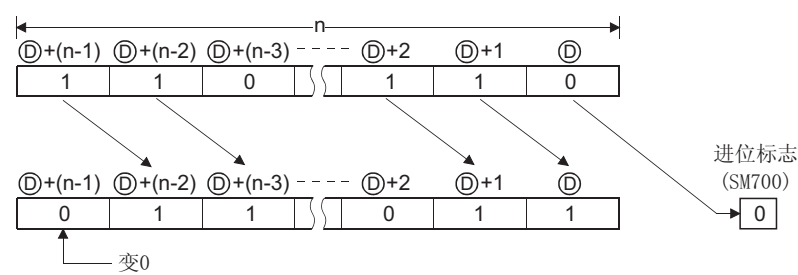
[设定数据]

设定数据	含义	数据类型
①	要移位的软元件的第一个软元件号	位
n	将移位的软元件数。	BIN 16 位

[功能]

BSFR

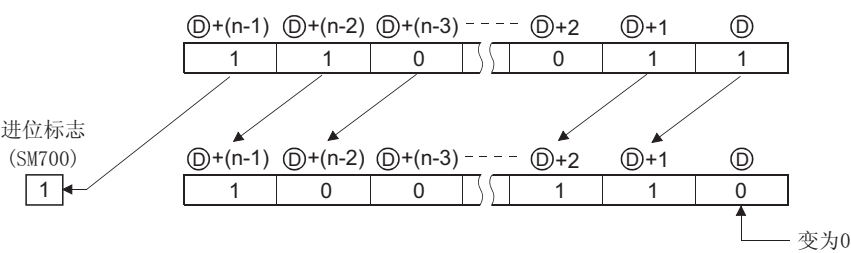
(1) 将从①指定的软元件开始的 n 点数据右移 1 位。



(2) ①+(n-1) 指定的软元件变为 0。

BSFL

(1) 将从①指定的软元件开始的 n 点数据左移 1 位。



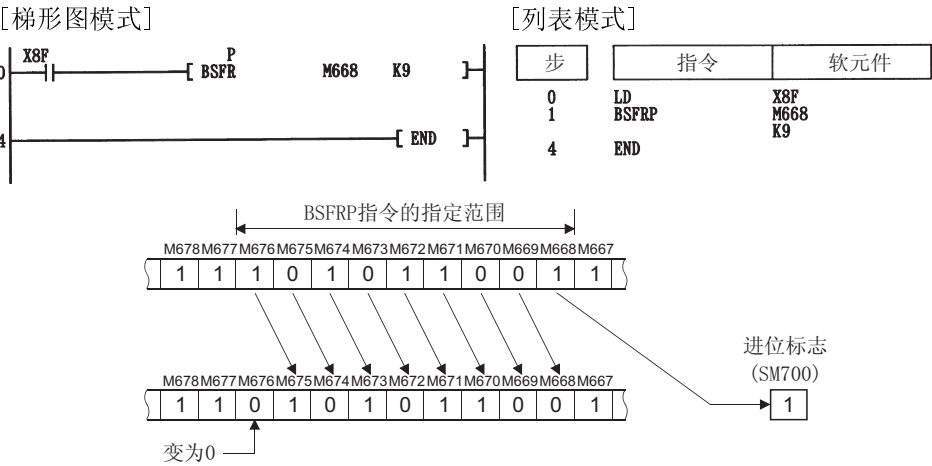
(2) ⑤指定的软元件变为 0。

[运行错误]

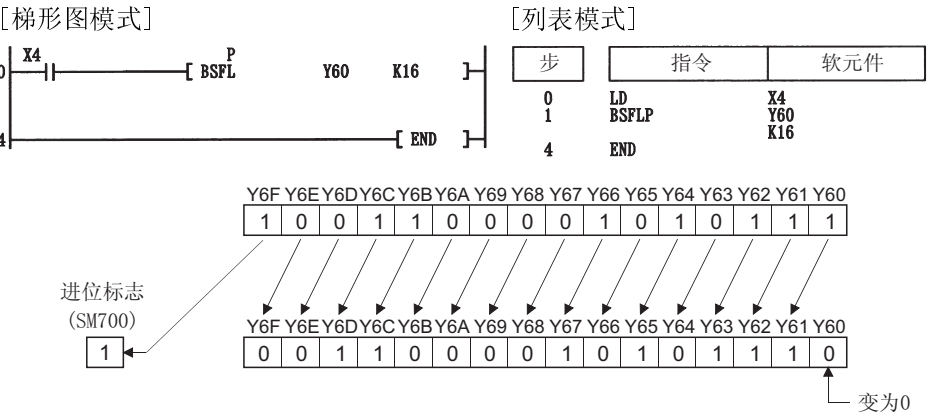
- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 ON，错误代码存储在 SD0 中。
- 从④软元件开始的 n 点数据软元件的范围超过了相关软元件。(错误代码:4101)

[程序示例]

(1) 当 X8F 为 ON 时，下列程序将 M668 到 M676 移位到右边。



(2) 当 X4 为 ON 时，下述程序将 Y60 到 Y6F 的输出移位到左边。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.3.3 n 字数据的 1 字左移或右移 (DSFR, DSFRP, DSFL, DSFLP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]K[]		特殊功能模 块 U[]V[]G[]	变址寄存器 Zn	常数 E	其它
	位	字		位	字				
①	—	○		—				—	
n	○	○		○				—	

[指令符号]	[执行条件]	□ 表示 DSFR/DSFL
DSFR, DSFL		
DSFRP, DSFLP		

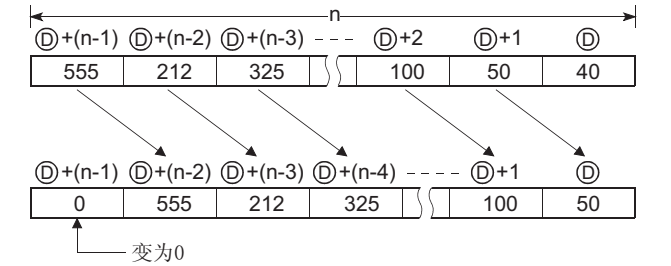
[设定数据]

设定数据	含义	数据类型
Ⓓ	要移位的软元件的第一个软元件号	BIN 16 位
n	要进行移位的软元件数	

[功能]

DSFR

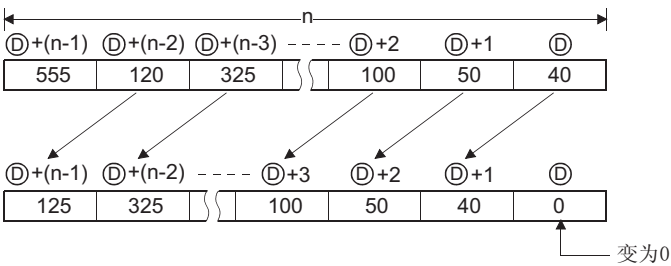
(1) 将从Ⓓ指定的软元件开始的 n 点数据右移 1 字。



(2) Ⓓ+(n-1) 指定的软元件变为 0。

DSFL

(1) 将从Ⓓ指定的软元件开始的 n 点数据左移 1 字。



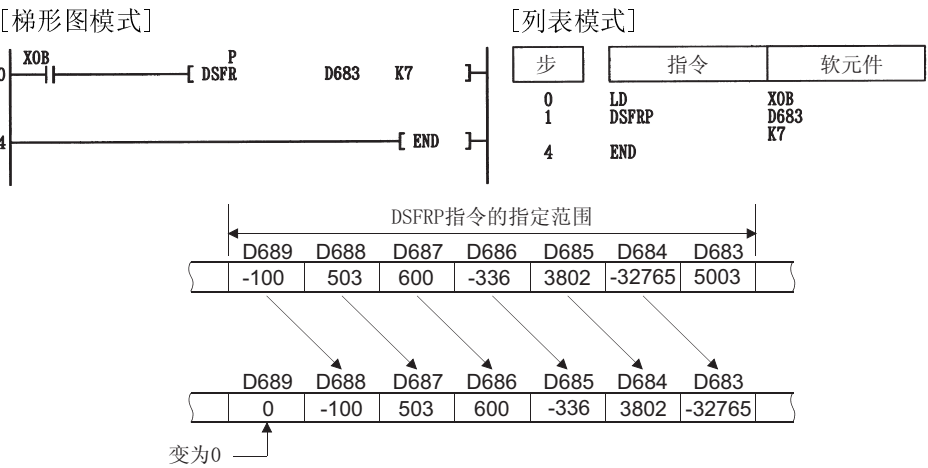
(2) ④指定的软元件变为 0。

[运行错误]

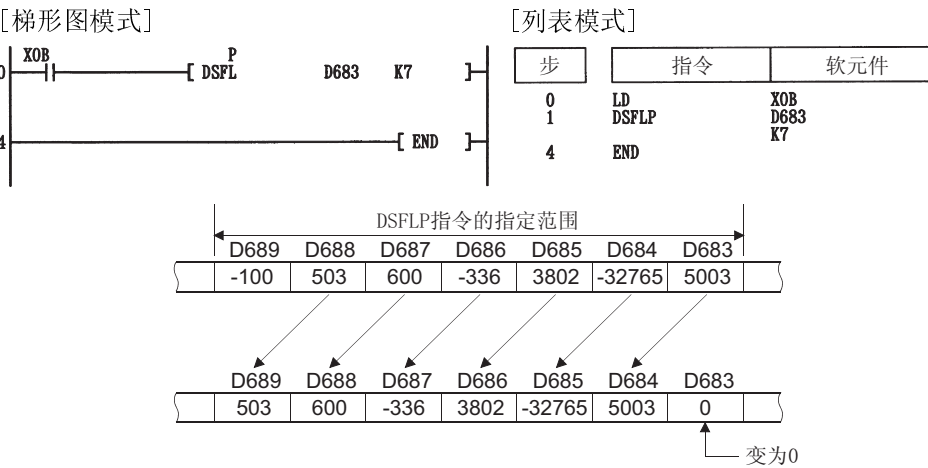
- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 从④软元件开始的 n 点软元件的范围超过了相关软元件。(错误代码:4101)

[程序示例]

(1) 当 XB 为 ON 时，下述程序将 D683 到 D689 的内容移位到右边。



(2) 当 XB 变为 ON 时，下述程序将 D683 到 D689 的内容移位到左边。

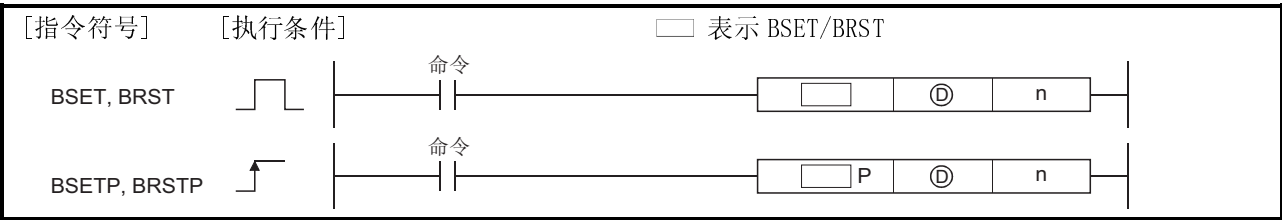


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.4 位处理指令

7.4.1 字软元件的位设定和复位(BSET, BSETP, BRST, BRSTP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]K[]		特殊功能模 块 U[]V[]G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
①	○						—	—	
n	○						○	—	



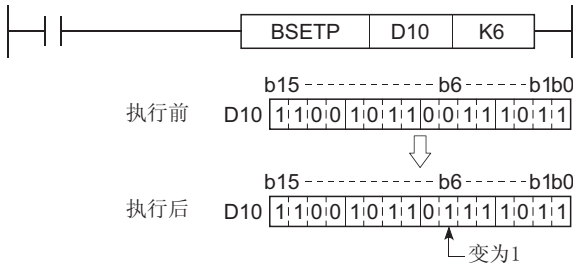
[设定数据]

设定数据	含义	数据类型
Ⓓ	进行置位或复位的软元件号	BIN 16 位
n	进行置位或复位的位号	

[功能]

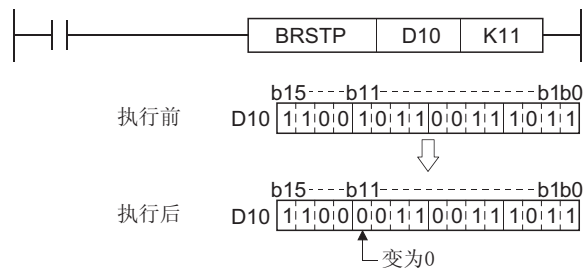
BSET

- (1) 对Ⓓ指定的字软元件的第 n 位进行置位(设置为“1”)。
- (2) n 值可以设定在 0 到 15 之间。
如果此值超过 15, 此运算将执行数据的低 4 位。



BRST

- (1) 将Ⓓ指定的字软元件的第 n 位复位到 0。
- (2) n 值可以设定在 0 到 15 之间。
如果此值超过 15, 此运算将执行数据低 4 位。

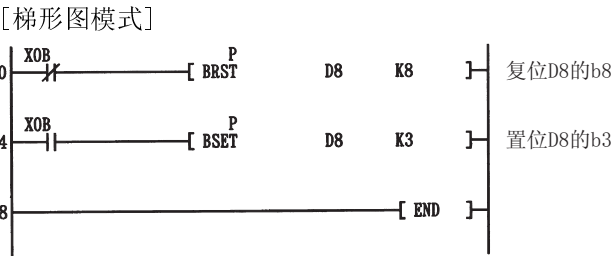


[运行错误]

(1) 没有与 BSET (P) 或 BRST (P) 指令相关的运行错误。

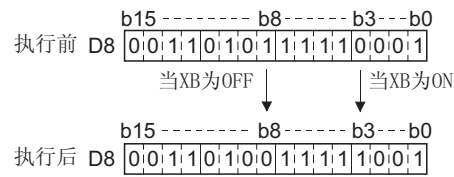
[程序示例]

(1) 当 XB 为 OFF 时，下列程序将 D8 的第 8 位 (b8) 复位到 0，当 XB 为 ON 时将 D8 的第 3 位 (b3) 置位到 1。



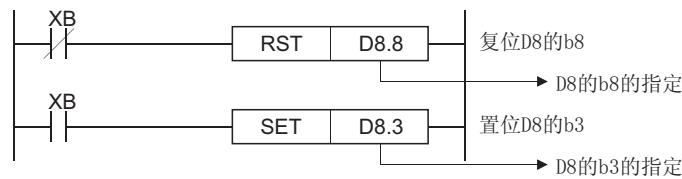
[列表模式]

步	指令	软元件
0	LDI	XB
1	BRSTP	D8 K8
4	LD	XB
5	BSETP	D8 K3
8	END	



备注

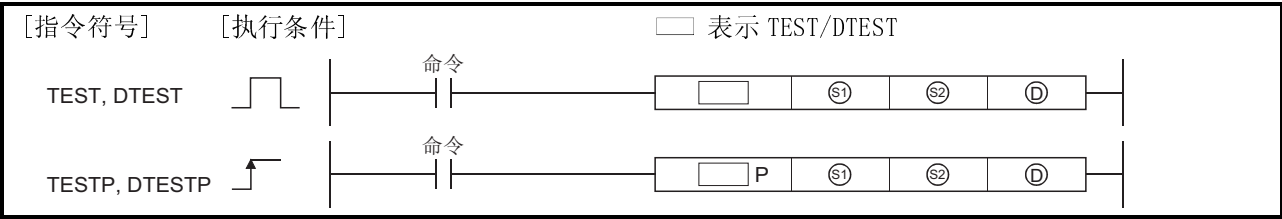
字软元件的置位或复位也可以通过字软元件的位指定来进行。
(关于字软元件的位指定请参考所使用的 CPU 模块的用户手册(功能说明, 编程基础)或 QnACPU 编程手册(编程基础))
如果使用了字软元件的位指定, 编程示例(1)处理过程可按如下进行。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.4.2 位测试(TEST, TESTP, DTEST, DTESTP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[N]C		特殊功能模 块 U[N]G	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
①	○						○	—	—
②	○						○	○	—
③	○						—	—	—



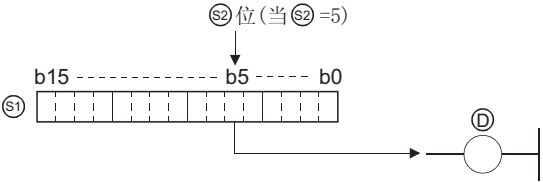
[设定数据]

设定数据	含义	数据类型
①	存储被提取的位数据的软元件数	字
②	被提取的位数据的位置 (0 到 15/0 到 31)	字
③	存储被提取的位数据的位软元件数	位

[功能]

TEST

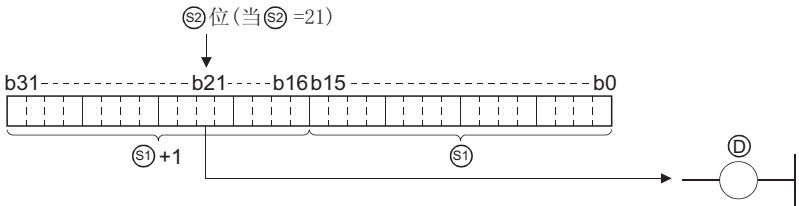
- (1) ①指定的字软元件内, 在②指定的位置取出位数据, 并且将它写到③指定的位软元件。
- (2) 当相关位为“0”时, ③指定的位软元件为 0FF, 当相关位为“1”时, ③指定的位软元件为 0N。
- (3) ②指定的位置表示了在一个字数据块(0 到 15)里单个位的位置。
当在②指定了 16 或更大的数时, 则目标是 n 除以 16 的余数指定的位置的位数据。
例如, 当 n=18 时, 因为 18 除以 16 的余数为 2, 所以目标为在 b2 处的数据。



DTEST

- (1) 在①或①+1 指定的 2 字的软元件内指定的位置取出位数据, 并且将它写到③指定的位软元件。
- (2) 当相关位为“0”时, ③指定的位软元件为 0FF, 当相关位为“1”时, ③指定的位软元件为 0N。
- (3) ②指定的位置表示了在一个 2 字数据块(0 到 31)里单个位的位置

当在⑤指定了 32 或更大的数时，则目标是 n 除以 32 的余数指定位置的位数据。
 例如，当 n=34 时，因为 34 除以 32 的余数为 2，所以目标为在 b2 的数据。



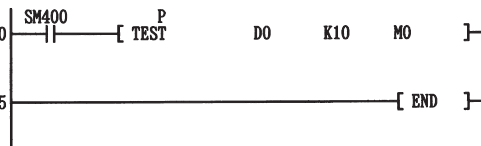
[运行错误]

(1) 没有与 TEST (P) 或 DTEST (P) 指令相关的运行错误。

[程序示例]

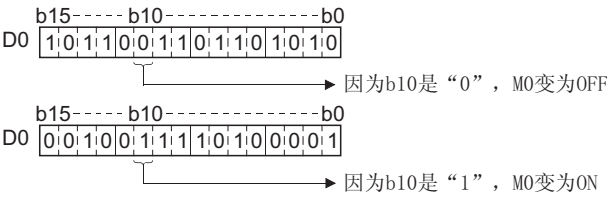
(1) 根据在 1 字数数据块 (D0) 里的第 10 位的状态，下列程序将 M0 变为 ON 或 OFF。

[梯形图模式]



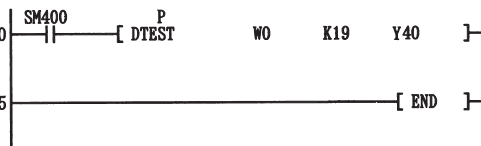
[列表模式]

步	指令	软元件
0	LD	SM400
1	TESTP	D0 K10 M0
5	END	



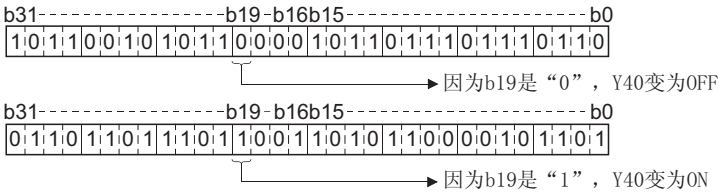
(2) 根据在 W0 和 W1 的 2 字数数据块 (D0) 里的第 19 位的状态，下列程序将 Y40 变为 ON 或 OFF。

[梯形图模式]



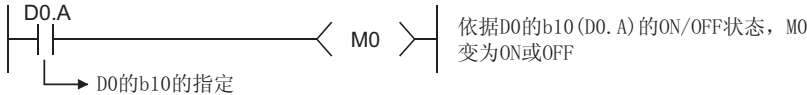
[列表模式]

步	指令	软元件
0	LD	SM400
1	DTESTP	W0 K19 Y40
5	END	



备注

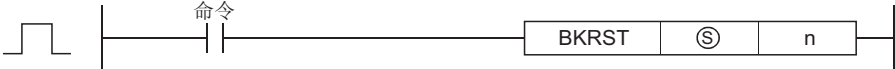
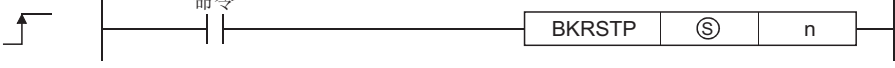
可以像使用字软元件的位指定那样使用位测试指令写程序。
 如果示例 (1) 中的程序改变为使用字软元件的位指定，它将显示如下。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.4.3 位软元件的成批复位(BKRST, BKRSTP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
⑤	○			—				—	
n	○			○				—	

[指令符号]	[执行条件]
BKRST	
BKRSTP	

[设定数据]

设定数据	含义	数据类型
⑤	被复位的软元件的第一个软元件号	位
n	被复位的软元件数	BIN 16 位

[功能]

(1) 从⑤指定的位软元件中复位 n 位位软元件。

软元件	状态
报警器 (F)	<ul style="list-style-type: none">● 将⑤指定的报警器 (F) 号里的软元件的 n 点置为 OFF。● 将报警器号 SD64 到 SD79 的数据置为 OFF 并且将剩余的数据向前压缩。● 将存储在 SD64 到 SD79 里的报警器号存储在 SD63 中。
(T) 定时器 (C) 计数器	● 将从⑤指定的定时器 (T) 或计数器 (C) 开始的 n 点当前值设置成 0，并将线圈触点置为 OFF。
其它位软元件	● 将⑤指定的软元件的线圈或触点 n 点置为 OFF。

(2) 如果指定的软元件是 OFF，则软元件状态将不会改变。

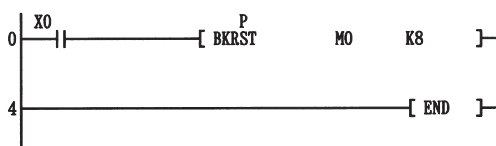
[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变 ON，并且错误代码存储在 SD0 中。
- 从⑤软元件开始的 n 点范围超过相关软元件。(错误代码:4101)

[程序示例]

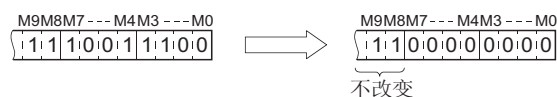
(1) 当 X0 变 ON 时，下列程序将 M0 到 M7 的软元件置为 OFF。

[梯形图模式]



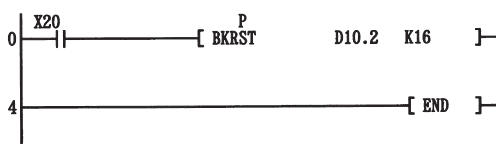
[列表模式]

步	指令	软元件
0	LD	X0
1	BKRSTP	M0 K8
4	END	



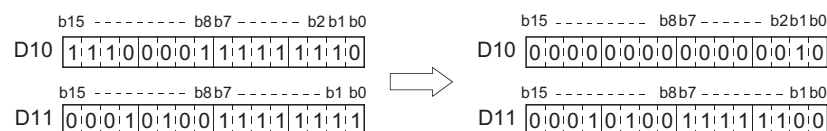
(2) 当 X20 变 ON 时，下列程序将 D10 的第二位 (b2) 到 D11 的第一位 (b1) 的数据置为 0。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X20
1	BKRSTP	D10.2 K16
4	END	

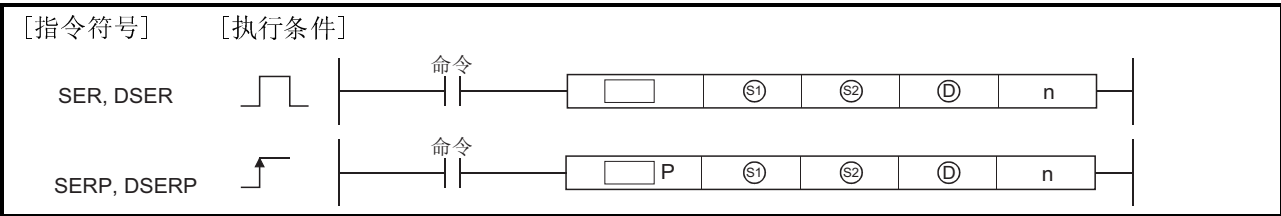


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.5 数据处理指令

7.5.1 16 位和 32 位数据搜索 (SER, SERP, DSER, DSERP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[] []		特殊功能模 块 U[] [G]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				U
①	○	○		○	○		○	—	
②	—	○		—	—		—	—	
③	—	○		—	○		—	—	
n	○	○		○	○		○	—	



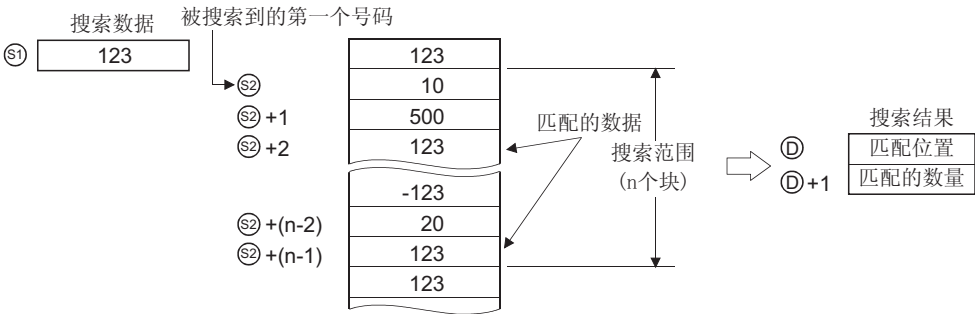
[设定数据]

设定数据	含义	数据类型
①	搜索数据或存储了搜索数据的软元件的第一个软元件号	BIN 16/32 位
②	搜索目标的数据, 或存储此类数据的第一个软元件。	BIN 16 位
③	将存储搜索结果的软元件的第一个软元件号	
n	搜索次数	

[功能]

SER

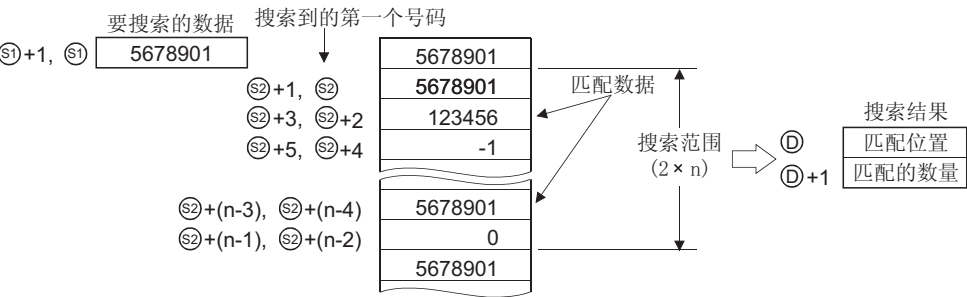
- (1) 将①指定的软元件的 16 位数据作为入口代码来搜索从②指定的软元件的 16 位数据开始的 n 个块。
- 与入口代码匹配的号码存储在③+1 指定的软元件中, 并且从②开始, 发现第一个匹配的软元件的点数的相关值存储在③指定的软元件中。



- (2) 如果 n 为 0 或负，则不进行处理。
- (3) 如果在搜索过程中没有发现匹配数据，则⑩和⑩+1 指定的软元件变为“0”。
- (4) 如果搜索的数据是按升序排列的，则在 SM702*为 ON 时可以通过使用带二分法搜索方法的搜索来提高处理速度。
如果搜索的数据不时按升序排列，则当 SM702 变 ON 时不能得到正确的搜索结果。

DSER

- (1) 将⑨+1, ⑨指定的软元件的 32 位数据作为搜索的入口代码，并且对从⑩(2×n 点)指定的软元件开始的 32 位数据单元里的 n 点进行搜索。
与入口代码的匹配数存储在⑩+1 指定的软元件中，并且从⑩开始，发现第一个匹配的软元件的点数的相关值存储在⑩指定的软元件中。



- (2) 如果 n 为 0 或负，则不进行处理。
- (3) 如果在搜索过程中没有发现匹配数，则⑩和⑩+1 指定的软元件变为“0”。

[运行错误]

- (1) 在下列情况下发生运行错误，则错误标志(SM0)变 ON，并且错误代码存储在 SD0 中。
- 从⑩软元件开始的 n 点数据的位置超过指定的软元件范围。 (错误代码:4101)
- (2) 关于在变址修改过程中的错误信息请参考 3.6 章节。

备注

- *:SM702 是设置搜索方法的特殊继电器。
- SM702OFF: 连续搜索
 - SM702ON: 二分法搜索方法。
- 被搜索的数据与要搜索的数据的比较方法，从数据头开始。
- 被搜索的数据里指定位置的数据与要搜索的数据的进行比较，此方法决定搜索目标存在于此位置的哪一侧，不断重复此步骤来缩小搜索范围，直到找到搜索目标。

[程序示例]

(1) 当 X20 为 ON 时，下述程序将搜索 D100 到 D105 的 D0 内容，并将搜索结果存储到 W0 和 W1。

[梯形图模式]

0

X20

LD

6

SER P

D0

D100

W0

K6

END

[列表模式]

步	指令	软元件
0	LD	X20
1	SERP	D0
		D100
		W0
		K6
6	END	

要搜索的数据

D0

123

被搜索的数据

D100	500
D101	123
D102	300
D103	123
D104	32000
D105	122

搜索结果

W0	2	匹配位置
W1	2	匹配数

(2) 当 X20 为 ON 时，下述程序将在 D100 到 D111 中搜索 D11 和 D10 的内容，并将搜索结果存储到 W0 和 W1。

[梯形图模式]

0

X20

LD

6

DSERP P

D10

D100

W0

K6

END

[列表模式]

步	指令	软元件
0	LD	X20
1	DSERP	D10
		D100
		W0
		K6
6	END	

要搜索的数据

D11, D10

56789051

被搜索的数据

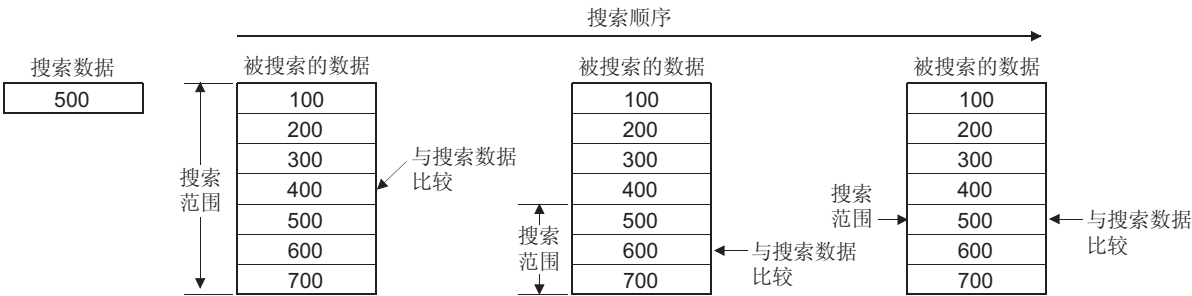
D101, D100	200000
D103, D102	56789051
D105, D104	56789051
D107, D106	-30000
D109, D108	56789051
D111, D110	30000

搜索结果

W0	2	匹配位置
W1	3	匹配数

备注

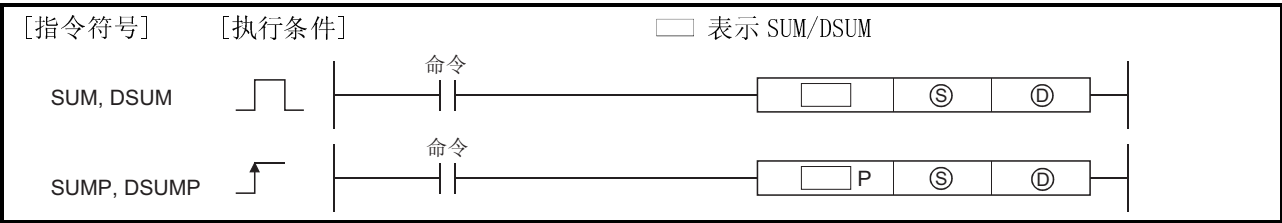
二分法搜索方法：将被搜索数据里中间的数据与搜索数据进行比较，根据两者的大小关系来决定下一步搜索的范围，然后重复进行这种比较，直到找到搜索数据。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.5.2 16 位和 32 位数据检查 (SUM, SUMP, DSUM, DSUMP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	



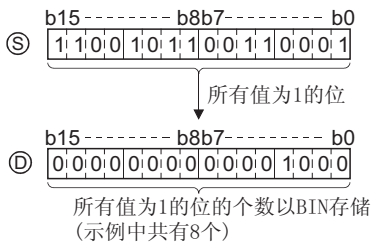
[设定数据]

设定数据	含义	数据类型
Ⓔ	计数所有是 1 的位的总数的软元件的第一个软元件号	BIN 16/32 位
Ⓕ	存储位总数的软元件的第一个软元件号	

[功能]

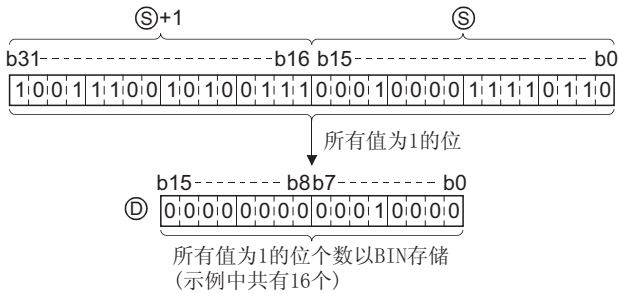
SUM

在Ⓔ指定的软元件里的 16 位数据中, 将所有值为 1 的位的个数存储在Ⓕ指定的软元件中。



DSUM

在Ⓔ指定的软元件里的 32 位数据中, 将所有值为 1 的位的总数存储在Ⓕ指定的软元件中。

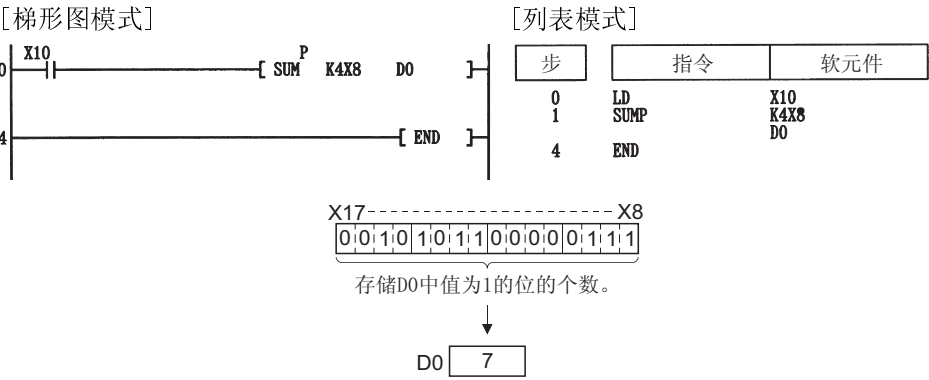


[运行错误]

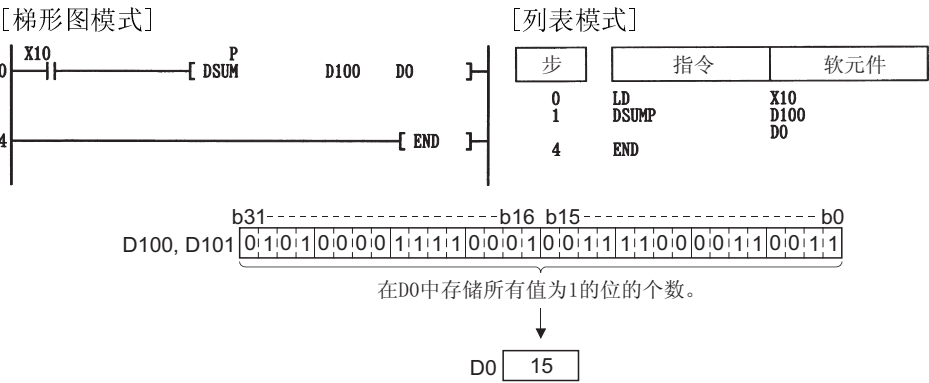
(1) 没有与 SUM(P)或 DSUM(P)指令相关的运行错误。

[程序示例]

(1) 当 X10 为 ON 时，下列程序将 X8 到 X17 里为 ON 的位的位数存储到 D0 中。





(2) 当 X10 为 ON 时，下列程序将 D100 和 D101 里为 ON 的位的个数存储到 D0 中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.5.3 8 位到 256 位的解码(DECO, DECOP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
Ⓔ	○			○				○	—
Ⓕ	○			—				—	—
n	○			○				○	—

[指令符号]	[执行条件]
DECO	
DECOP	

[设定数据]

设定数据	含义	数据类型
Ⓔ	解码的数据或储存了解码数据的软元件	BIN 16/32 位
Ⓕ	存储解码结果的软元件的第一个软元件号	软元件名
n	有效位的长度(1 到 8)，0: 不处理	BIN 16 位

[功能]

DECO

- (1) 解码Ⓔ指定的软元件的低 n 位，并将结果存储在Ⓕ指定的软元件的 2ⁿ 位的位置。
- (2) n 值可以在 1 到 8 之间指定。
- (3) 如果 n=0 则不进行处理，在Ⓕ指定的软元件里 2ⁿ 位没有改变。
- (4) 位软元件作为 1 位，字软元件作为 16 位。

[运行错误]

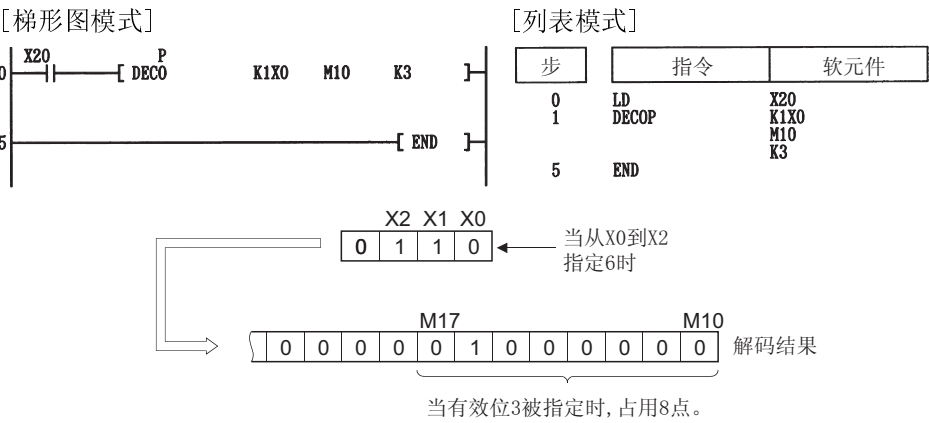
- (1) 在下列情况下发生运行错误，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- n 值不在 0 到 8 范围之内。

(错误代码:4100)
- ⑤里 2ⁿ 位范围超过了相关软元件范围。

(错误代码:4101)

[程序示例]

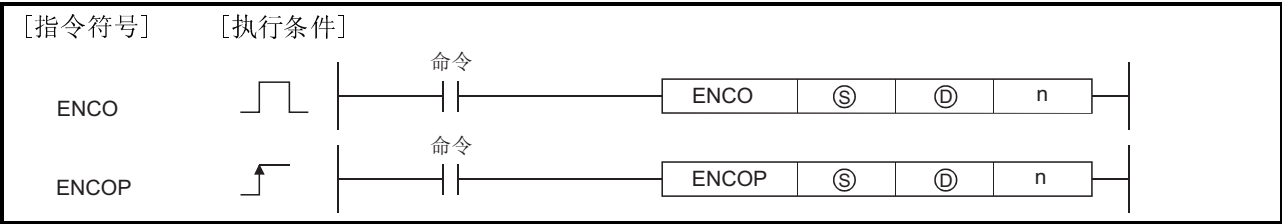
(1) 当 X20 为 ON 时，下列程序将 X0 开始的 3 位数据解码并将结果存储在 M10 中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.5.4 将 256 位编码为 8 位 (ENCO, ENCOP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[] []		特殊功能模 块 U[] [G]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
Ⓔ	○			—				○	—
Ⓕ	○			○				—	—
n	○			○				○	—



[设定数据]

设定数据	含义	数据类型
Ⓢ	编码的数据，或存储编码数据软元件的第一个软元件号	软元件名
ⓓ	将存储编码结果的软元件号。	BIN 16 位
n	有效位长度(1 到 8)，0:无处理。	

[功能]

ENCO

- (1) 将Ⓢ里的 2ⁿ 位数据编码并存储在ⓓ里。
- (2) n 的有效设置为 1 到 8。
- (3) 如果 n=0，将没有运算操作，并且ⓓ的内容不改变。
- (4) 位软元件作为 1 位处理，字软元件作为 16 位处理。
- (5) 如果有几位数据都 1，将在最高位进行处理。

[运行错误]

- (1) 在下列情况下发生运行错误，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- n 值不在 0 到 8 范围内。

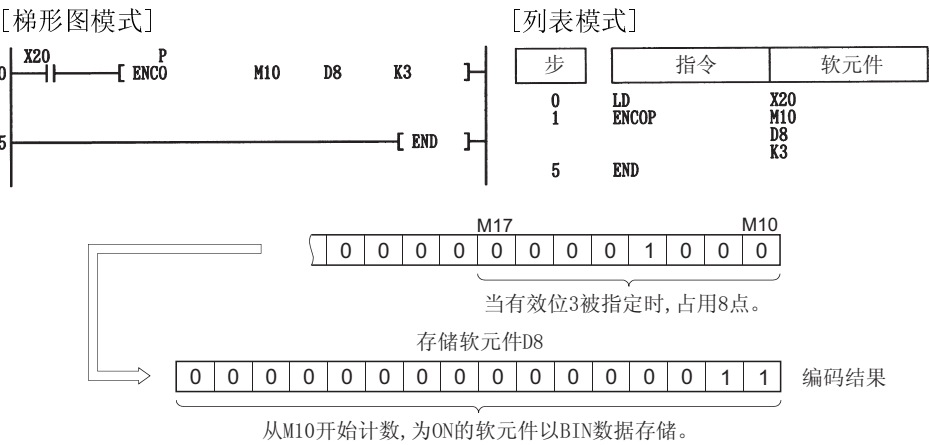
(错误代码:4100)
- 从⑤开始的 2ⁿ 位的值超过 0 到 8 范围。

(错误代码:4101)
- 从⑤开始的所有 2ⁿ 位数据为 “0” 。

(错误代码:4100)

[程序示例]

- (1) 当 X20 为 ON 时，下述程序将 M10 开始的 3 位进行编码，并将结果存储在 D8 中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.5.5 7 段解码 (SEG, SEGP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	

[指令符号]	[执行条件]
SEG	
SEGP	

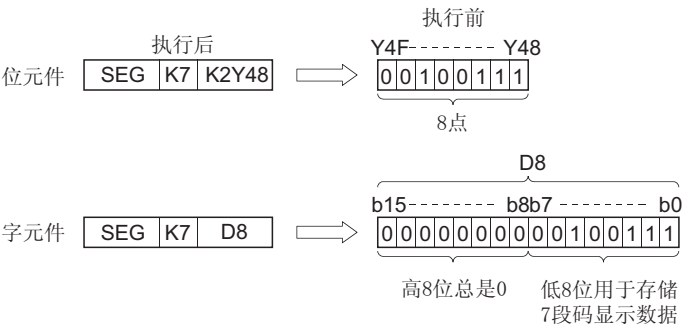
[设定数据]

设定数据	含义	数据类型
⑤	解码的数据或存储解码数据的软元件的第一个软元件号	BIN 16 位
⑥	存储解码结果的软元件的第一个软元件号。	

[功能]

SEG

- (1) 将⑤低 4 位指定的 0 到 F 数据解码成 7 段显示数据，并存储到⑥中。
- (2) 如果⑥是一个位软元件，指示存储 7 段显示数据的软元件的第一个软元件号。如果⑥是一个字软元件，则指示存储数据的软元件数。
- (3) 对位软元件和字软元件，存储过程如下：



[运行错误]

- (1) 没有与 SEG (P) 指令相关的运行错误。

7-段解码显示

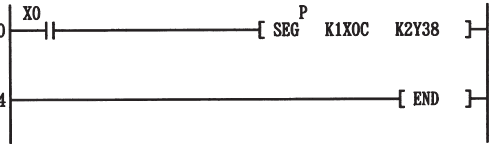
Ⓢ		7 段码配置	ⓐ							显示数据
十六进制	位状态		B7	B6	B5	B4	B3	B2	B1	
0	0000		0	0	1	1	1	1	1	0
1	0001		0	0	0	0	0	1	1	1
2	0010		0	1	0	1	1	0	1	2
3	0011		0	1	0	0	1	1	1	3
4	0100		0	1	1	0	0	1	1	4
5	0101		0	1	1	0	1	1	0	5
6	0110		0	1	1	1	1	1	0	6
7	0111		0	0	1	0	0	1	1	7
8	1000		0	1	1	1	1	1	1	8
9	1001		0	1	1	0	1	1	1	9
A	1010		0	1	1	1	0	1	1	A
B	1011		0	1	1	1	1	1	0	b
C	1100		0	0	1	1	1	0	0	c
D	1101		0	1	0	1	1	1	1	d
E	1110		0	1	1	1	1	0	0	e
F	1111		0	1	1	1	0	0	1	F

位元件第一位
字元件的最低位

[程序示例]

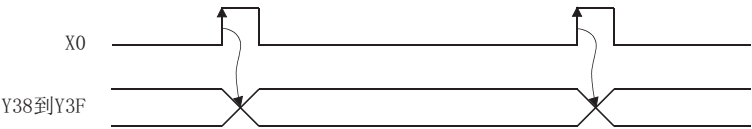
(1) 当 X0 为 ON 时，下列程序将 Xc 到 XF 的数据转换成 7 段码显示数据并输出到 Y38 到 Y3F 中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	SEGP	K1XOC K2Y38
4	END	



* 从Y38到Y3F中的数据在下一个数据被输出之前不会改变。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.5.6 16 位数据的 4 位分组 (DIS, DISP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
Ⓢ	○	○				○	—
Ⓓ	—	○				—	—
n	○	○				○	—

[指令符号]	[执行条件]
DIS	
DISP	

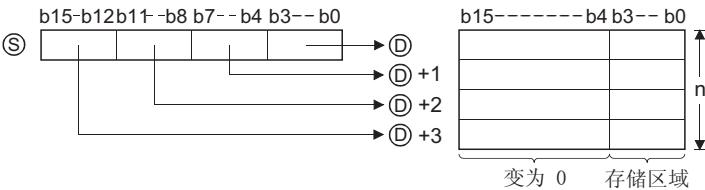
[设定数据]

设定数据	含义	数据类型
Ⓢ	存储要分离的数据的软元件的第一个软元件号	BIN 16 位
Ⓓ	存储已分离的数据的软元件的第一个软元件号	
n	分离数 (1 到 4)：如果分离数为 0，则不执行任何处理	

[功能]

DIS

(1) 将Ⓢ指定的 16 位数据的低 n 位数字 (1 位数字为 4 位) 存储在从Ⓓ指定的软元件开始的 n 点的低 4 位。



- (2) 从Ⓓ指定的软元件开始的 n 点的高 12 位变为 0。
- (3) 可以指定给 n 的值的范围是 1 到 4。
- (4) 如果 n=0，则不执行任何运算，并且从Ⓓ开始的 n 点中的内容将保持不变。

[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 从⑤开始的 n 点范围超出了相关软元件。

(错误代码:4101)

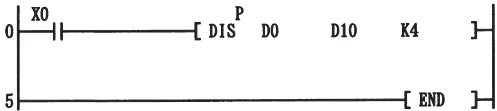
● n 的值超出了 0 到 4 的范围。

(错误代码:4100)

[程序示例]

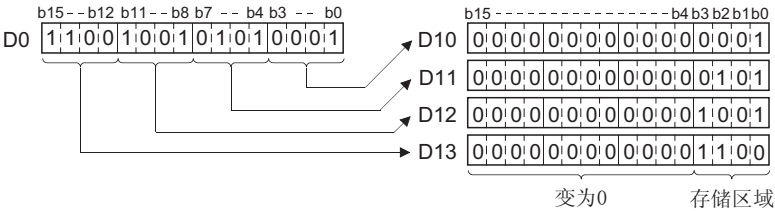
(1) 当 X0 为 ON 时，以下程序将 D0 中的 16 位数据分离成 4 位一组，并存储在 D10 到 D13 中。

[梯形图模式]



[列表模式]

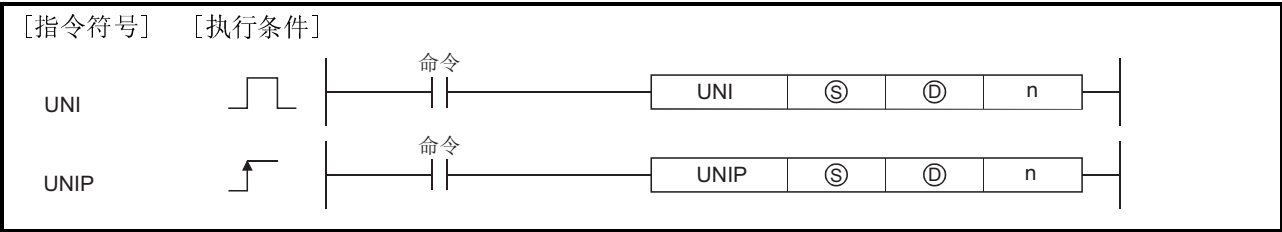
步	指令	软元件
0	LD	X0
1	DISP	D0 D10 K4
5	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.5.7 16 位数据的 4 位链接 (UNI, 与 UNIP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
Ⓔ	—	○		—				—	—
Ⓕ	○	○		○				—	—
n	○	○		○				○	—



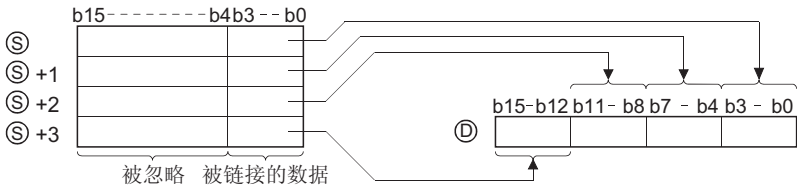
[设定数据]

设定数据	含义	数据类型
Ⓔ	存储要链接的数据的软元件的第一个软元件号	BIN 16 位
Ⓕ	存储已链接的数据的软元件的第一个软元件号	
n	链接数 (1 到 4): 如果链接数为 0, 则不执行任何处理	

[功能]

UNI

(1) 将从Ⓔ指定的软元件开始的 n 点中的 16 位数据的低 4 位链接到Ⓕ指定的 16 位软元件。



(2) Ⓕ指定的软元件的高位 (4 到 n) 数字的各位变为 0。

(3) 可以指定给 n 的值的范围是 1 到 4。

(4) 如果 n=0, 则不执行任何运算, 并且软元件Ⓕ中的内容将保持不变。

[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 从⑤开始的 n 点范围超出了相关软元件。

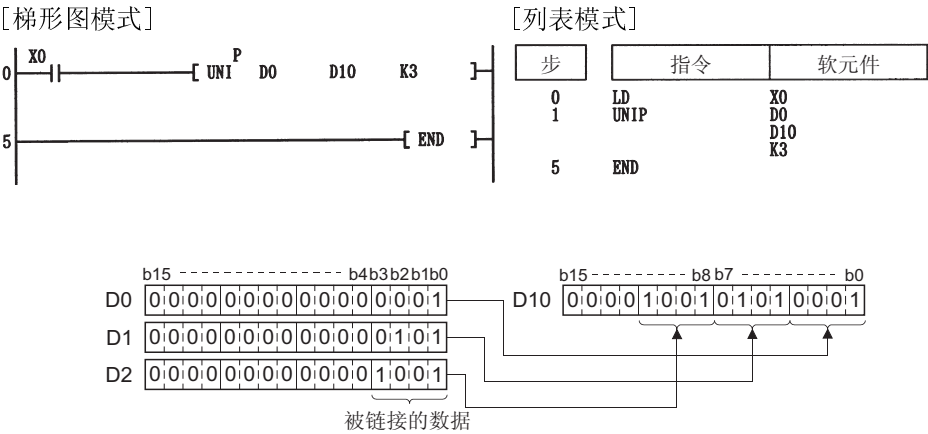
(错误代码:4101)

● n 的值超出了范围 0 到 4。

(错误代码:4100)

[程序示例]

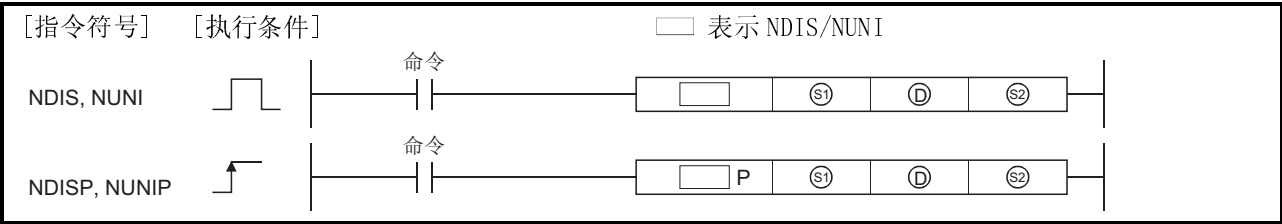
(1) 当 X0 为 ON 时，以下程序链接 D0 至 D2 中数据的低 4 位，并将它们存储在 D10 中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.5.8 随机数据的分离或链接 (NDIS, NDISP, NUNI, 与 NUNIP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [J][K]		特殊功能模 块 [L][M]	变址寄存器 Zn
	位	字		位	字		常量 K, H
[S1]	—	○					—
[D]	—	○					—
[S2]	—	○					—



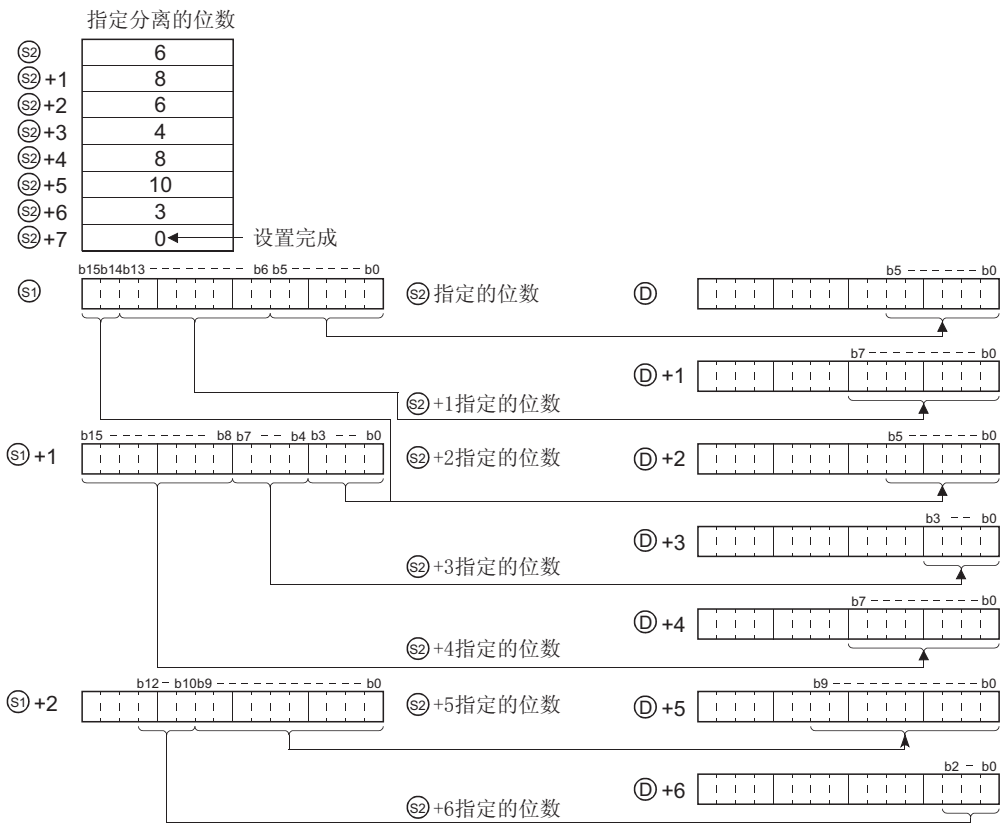
[设定数据]

设定数据	含义	数据类型
[S1]	存储要分离或链接的数据的软元件的第一个软元件号	BIN 16 位
[D]	存储已分离或链接的数据的软元件的第一个软元件号	
[S2]	存储进行分离或链接时将使用的单位的软元件的第一个软元件号	

[功能]

NDIS

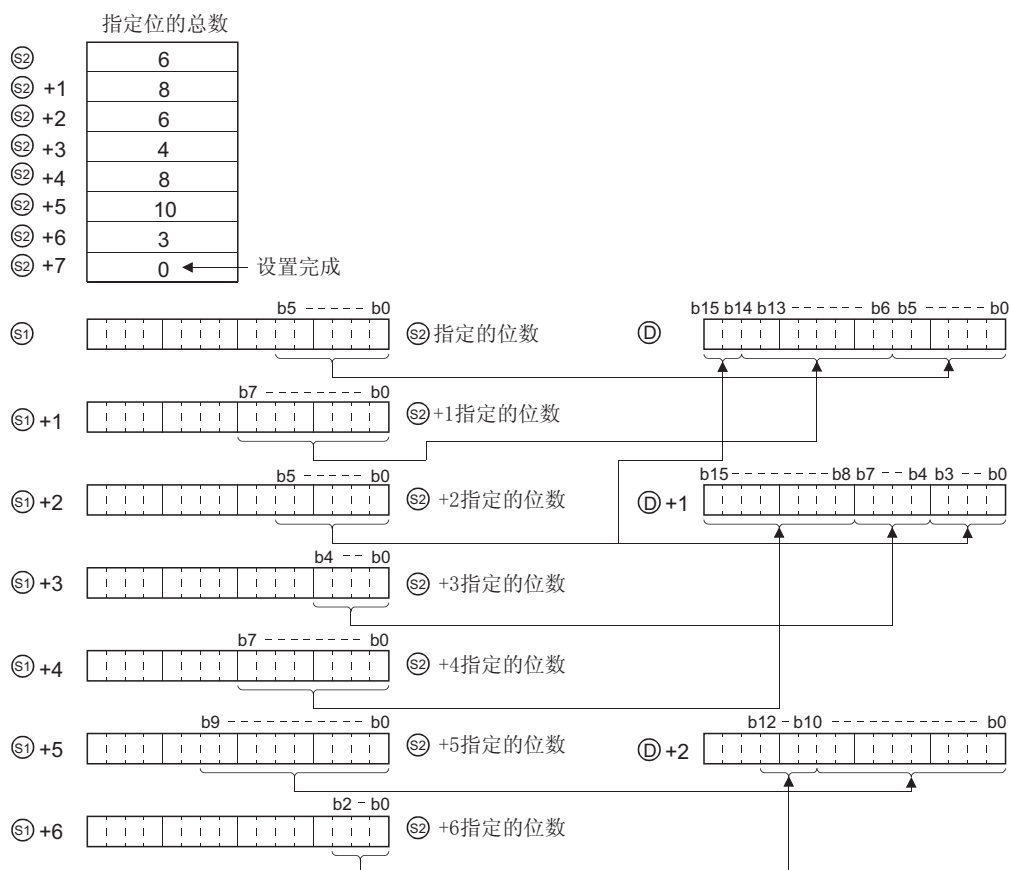
(1) 将从①指定的软元件开始的软元件中存储的数据分离成②指定的位数，并将该数据存储在从③指定的软元件开始的软元件中。



- (2) 可以指定给②的分离位数的范围是 1 到 16 位。
- (3) 从②指定的软元件号到存储“0”的软元件号之间的位都被作为分离位进行处理。

NUNI

(1) 按照⑤2指定的位数链接存储在从⑤1指定的软元件开始的软元件中的数据的单个位，并将它们存储在从⑤指定的软元件开始的软元件中。



(2) 可以指定给⑤2的链接位数的范围是 1 到 16。

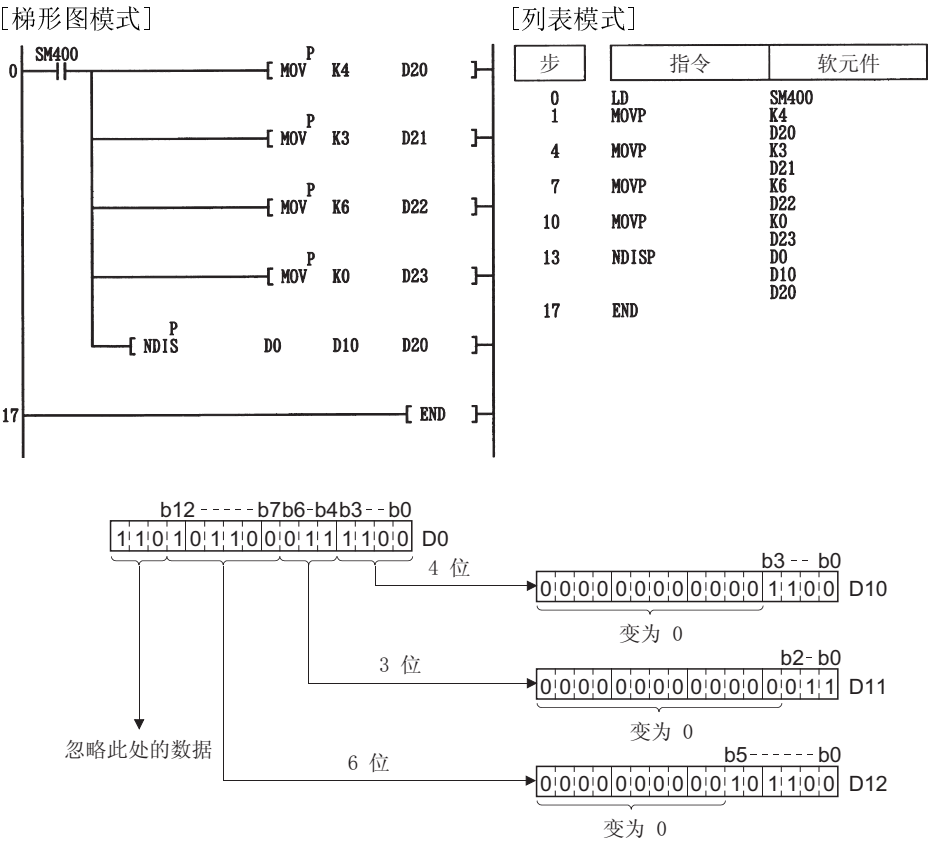
(3) 将从②指定的软元件号到存储“0”的软元件号之间要被链接的位的数目进行处理。

[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- ⑤②指定的分离或链接位数，或者⑤③或⑤④指定的软元件的使用范围超出了它们各自软元件的最后一个软元件号。(错误代码: 4101)
 - ⑤②指定的分离或链接位数不在范围 1 到 16 之内。(错误代码: 4100)

[程序示例]

- (1) 以下程序从 D0 中数据的低位开始分别分离出 4 位、3 位和 6 位数据，并将它们存储在 D10 到 D12 中。

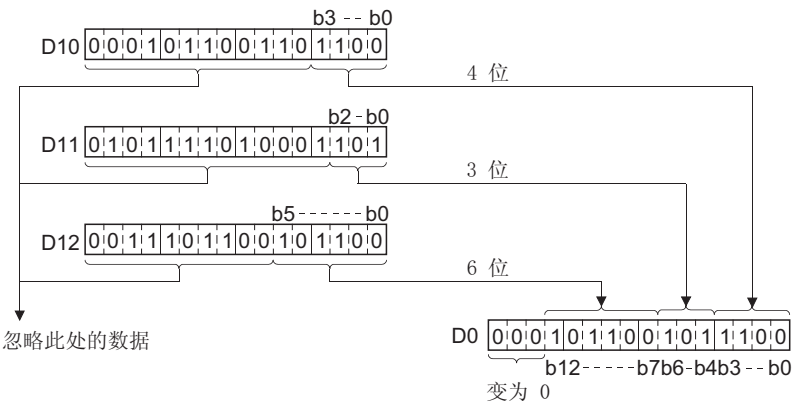


(2) 以下程序链接 D10 中数据的低 4 位，D11 中数据的低 3 位和 D12 中数据的低 6 位，并将它们存储在 D0 中。

[梯形图模式]

[列表模式]

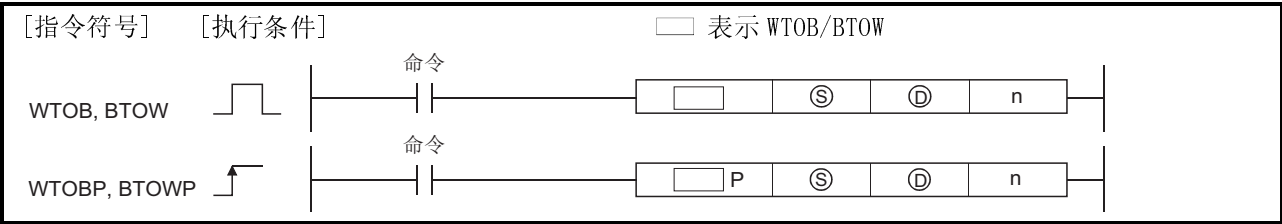
步	指令	软元件
0	LD	SM400
1	MOVP	K4 D20
4	MOVP	K3 D21
7	MOVP	K6 D22
10	MOVP	K0 D23
13	NDISP	D10 D0 D20
17	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.5.9 以字节为单位的数据分离和链接(WTOB, WTOBP, BTOW, 与 BTOWP)

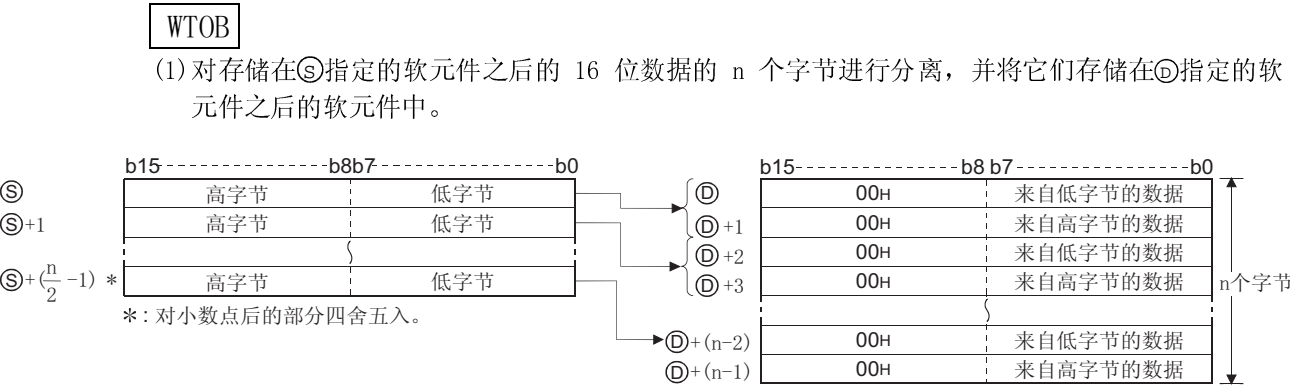
设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [J][K][L]		特殊功能模 块 [U][V][G][C]	变址寄存器 Zn
	位	字		位	字		
Ⓢ	—	○			—		—
Ⓓ	—	○			—		—
n	○	○			○		—



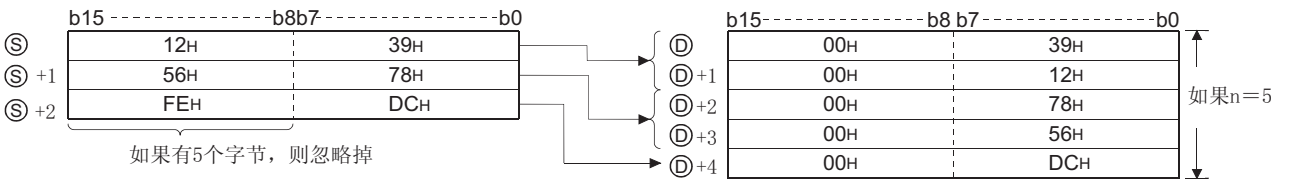
[设定数据]

设定数据	含义	数据类型
Ⓢ	存储需要以字节为单位进行分离或链接的数据的软元件的第一个软元件号	BIN 16 位
Ⓓ	存储以字节为单位进行分离或链接的结果的软元件的第一个软元件号	
n	要分离或链接的字节数	

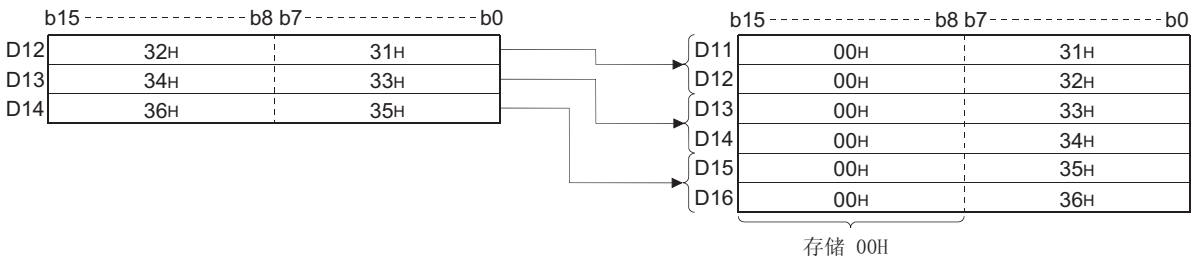
[功能]



例如, 如果 n=5, 则Ⓢ的低 8 位到(Ⓢ+2)中数据将被存储在Ⓓ到Ⓓ+4 中。

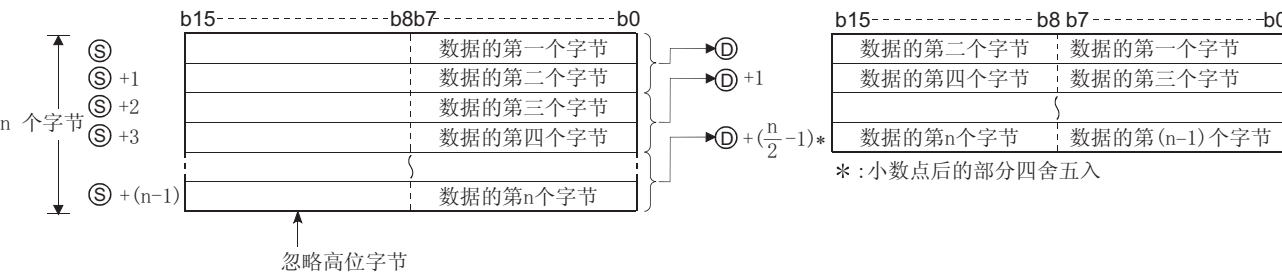


- (2)通过 n 对字节数的设定将自动确定⑤指定的 16 位数据的范围以及用于存储⑥指定的字节数据的软元件的范围。
- (3)当 n 指定的字节数为“0”时，不执行任何处理。
- (4)代码“00_H”将自动存储在⑥指定的字节存储软元件的高 8 位中。

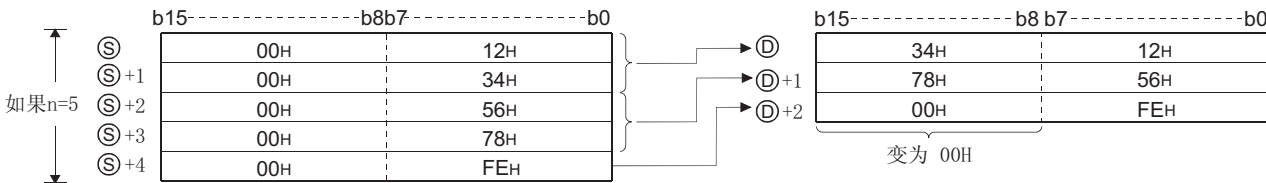


BTOW

- (1)以字为单位链接从⑤指定的软元件开始的软元件中的 n 字 16 位数据的低 8 位，并存储在从⑥指定的软元件开始的软元件中。
忽略从⑤指定的软元件开始的软元件中的 n 字数据的高 8 字节。
另外，如果 n 是奇数，那么存储数据的第 n 个字节的软元件的高 8 位将存储为 0。



例如，如果 n=5，则链接⑤到⑤+4 中数据的低 8 位，且存储在⑥到⑥+2 中。



- (2)通过 n 对字节数的设定将自动确定⑤指定的 16 位数据的范围以及用于存储⑥指定的字节数据的软元件的范围。
- (3)当 n 指定的字节数为“0”时，不执行任何处理。
- (4)忽略⑤指定的字节存储软元件的高 8 位，而使用其低 8 位。

- (5) 即使是在如下情况中，也会执行精确的处理：存储将要链接到的数据的软元件范围(㉔到㉔+(n-1))与存储被链接数据的软元件范围(㉕到㉕+(n-1))相互之间有重叠。
- 例如，当 D11 到 D16 的低 8 位需要存储在 D12 到 D14 中时，将发生下列情况：

	b15-----b8	b7-----b0		b15-----b8	b7-----b0
D11	00H	31H	}	D11	00H 31H
D12	00H	32H		D12	32H 31H
D13	00H	33H		D13	34H 33H
D14	00H	34H		D14	36H 35H
D15	00H	35H		D15	00H 35H
D16	00H	36H		D16	00H 36H

[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 存储在从㉔指定的软元件之后的软元件中，由 n 指定的字节数范围超出了相关软元件范围。
(错误代码:4101)
 - 存储在从㉕指定的软元件之后的软元件中，由 n 指定的字节数范围超出了相关软元件范围。
(错误代码:4101)

[程序示例]

- (1) 当 X0 为 ON 时，以下程序以字节为单位分离 D10 到 D12 中的数据，并存储在 D20 到 D25 中。

[梯形图模式]

```
graph LR
    X0((X0)) --> WOTOB[P WOTOB D10 D20 K6]
    WOTOB --> END[END]
```

[列表模式]

步	指令	软元件
0	LD	X0
1	WTOBP	D10 D20 K6
5	END	

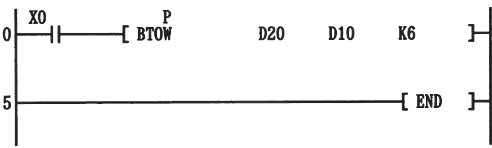
	b15-----b8	b7-----b0
D10	FDH	58H
D11	57H	E2H
D12	34H	44H

	b15-----b8	b7-----b0
D20	00H	58H
D21	00H	FDH
D22	00H	E2H
D23	00H	57H
D24	00H	44H
D25	00H	34H

6个字节

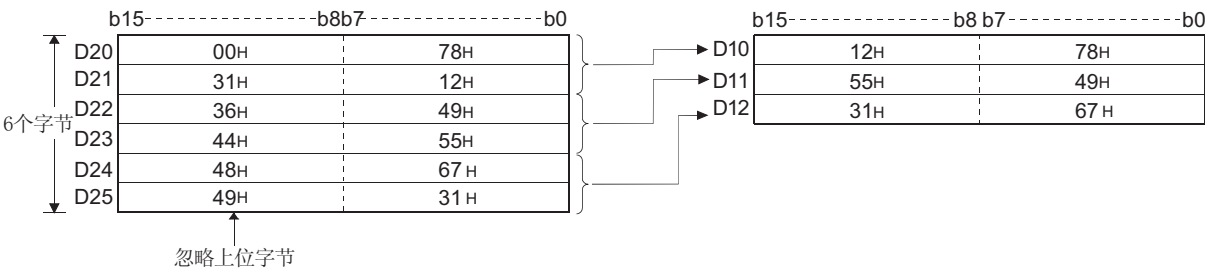
(2) 当 X0 为 ON 时，以下程序链接 D20 到 D25 中数据的低 8 位，并将结果存储在 D10 到 D12 中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	BTOWP	D20 D10 K6
5	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.5.10 16 位和 32 位数据的最大值查找 (MAX, MAXP, DMAX, 与 DMAXP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[K][L]		特殊功能模 块 U[1][6]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	—	○			—				—
⑥	—	○			—				—
n	○	○			○				—

[指令符号]	[执行条件]	□ 表示 MAX/DMAX	
MAX, DMAX			⑤ ⑥ n
MAXP, DMAXP			⑤ ⑥ n

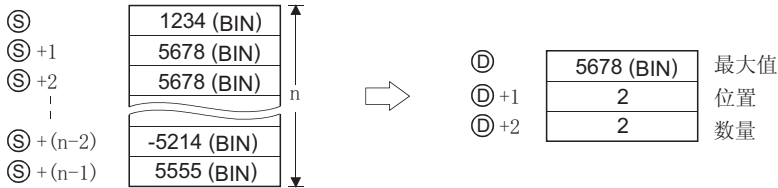
[设定数据]

设定数据	含义	数据类型
⑤	用于查找最大值的软元件的起始号	BIN 16/32 位
⑥	用于存储最大值查找结果的软元件的起始号	
n	查找的数据块数	BIN 16 位

[功能]

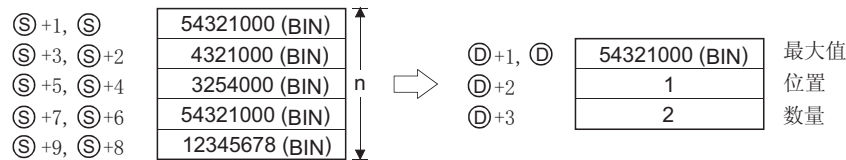
MAX

- (1) 从⑤指定的软元件开始的 n 块 16 位数据中查找最大值，并将该值存储在⑥指定的软元件中。
从⑤指定的软元件开始，查找到存储已发现的第一个最大值的软元件号，在⑥+1 中存储该软元件号距离软元件⑤的点数，在⑥+2 中存储最大值的数目。



DMAX

- (1) 从⑤指定的软元件开始的 n 块 32 位数据中查找最大值，并将该值存储在④和④+1 指定的软元件中。
- 从⑤指定的软元件开始，在查找到存储着已发现的第一个最大值的软元件号时，在④+2 中存储该软元件号距离软元件⑤的点数，在④+3 中存储最大值的数目。

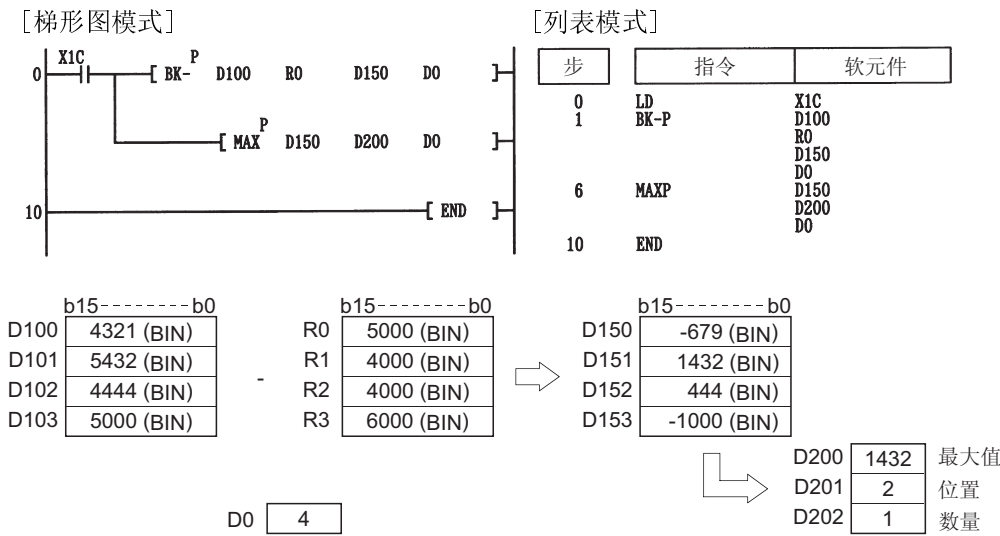


[运行错误]

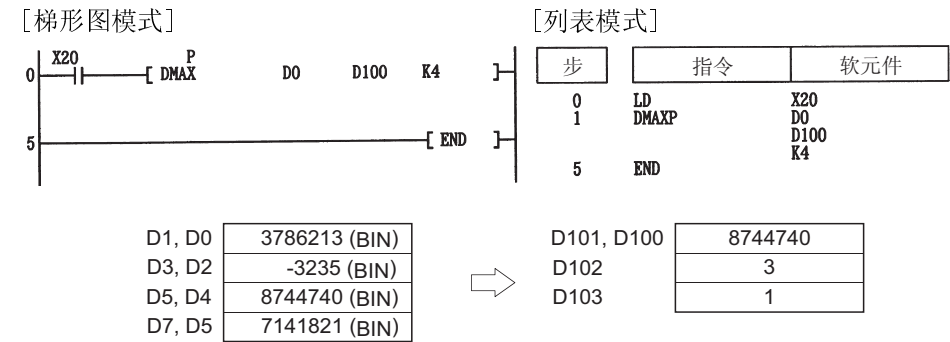
- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 从⑤指定的软元件开始的 n 点范围超出了相关软元件范围。 (错误代码:4101)

[程序示例]

- (1) 当 X1C 变为 ON 时，以下程序将从 D100 开始的软元件 (软元件的个数由存储在 D0 中的值指定) 中的数据中减去从 R0 开始的软元件 (软元件的个数由存储在 D0 中的值指定) 中的数据，再从该计算的结果中查找最大值，并将它存储在 D200 到 D202 中。



- (2) 当 X20 变为 ON 时，以下程序从 D0 到 D7 中的 32 位数据中查找最大值，并将它存储在 D100 到 D103 中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.5.11 查找 16 位和 32 位数据的最小值 (MIN, MINP, DMIN, 与 DMINP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[N]C		特殊功能模 块 U[J]G	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	—	○			—				—
⑥	—	○			—				—
n	○	○			○				—

[指令符号]	[执行条件]	□ 表示 MIN/DMIN			
MIN, DMIN		命令		⑤	⑥ n
MINP, DMINP		命令		P ⑤	⑥ n

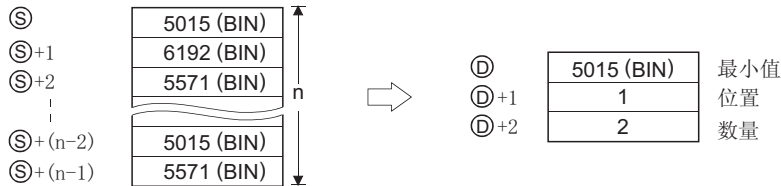
[设定数据]

设定数据	含义	数据类型
⑤	用于查找最小值的软元件的起始号	BIN 16/32 位
⑥	用于存储最小值查找结果的软元件的起始号	
n	查找的数据块数	BIN 16 位

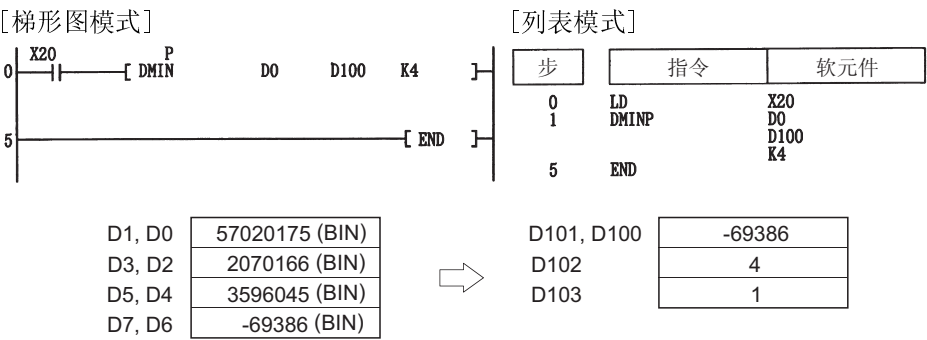
[功能]

MIN

- (1) 从⑤指定的软元件开始的 n 块 16 位 BIN 数据中查找最小值，并将该值存储在⑥指定的软元件中。
从⑤指定的软元件开始，在查找到存储着已发现的第一个最小值的软元件号时，在⑥+1 中存储该软元件号距离软元件⑤的点数，在⑥+2 中存储最小值的数目。



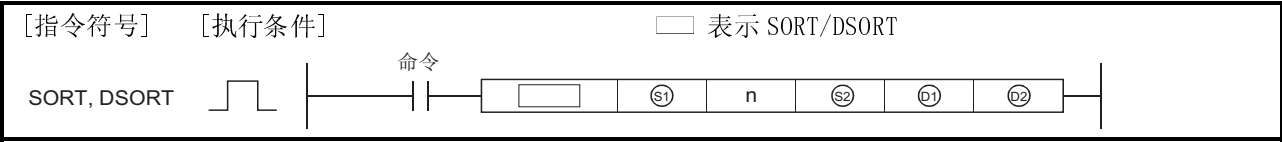
(2) 当 X20 变为 ON 时，以下程序从 D0 到 D7 中的 32 位数据中查找最小值，并将它存储在 D100 到 D103 中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.5.12 BIN 16 位和 32 位数据的排序运算 (SORT, DSORT)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
(S1)	—	○			—				—
n	○	○			○				—
(S2)	○	○			○				—
(D1)	○	—			—				—
(D2)	—	○			—				—



[设定数据]

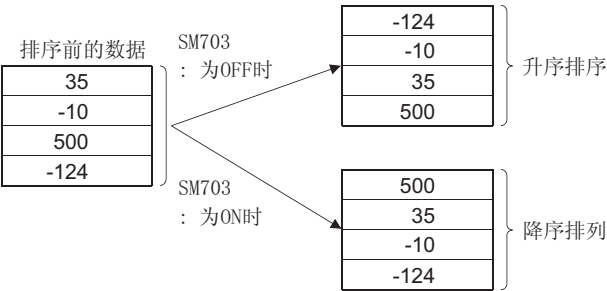
设定数据	含义	数据类型
(S1)	将进行排序的表的起始软元件号	BIN 16/32 位
n	要排序的数据块数	BIN 16 位
(S2)	在一次排序运算中用于比较的数据块数	BIN 16 位
(D1)	排序完成后被强制为 ON 的位软元件号	位
(D2)	系统使用的软元件	BIN 16 位

[功能]

- SORT

(1) 对从(S1)开始的 n 点 BIN 16 位数据按升序或者降序进行排序（重新排列数据）。

(2) 排序类型由 SM703 的状态 ON 或 OFF 指定：
 - 当 SM703 的状态为 OFF 时：升序排序
 - 当 SM703 的状态为 ON 时：降序排序

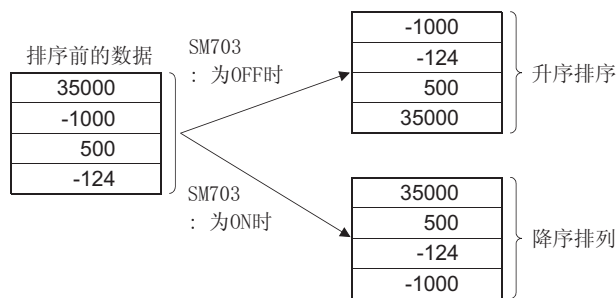


- (3) 用 SORT 指令执行排序需要若干个扫描周期。
在排序结束时共执行的最大次数除以执行一次排序运算所比较的数据块数 (由 $\textcircled{S2}$ 指定), 所得到的结果值就是到排序结束共执行的扫描周期数。(小数部分四舍五入)。
如果增大 $\textcircled{S2}$ 的值, 则在排序结束时扫描周期数减少, 但每次扫描的时间变长。
- (4) 在排序结束时共执行的最大次数应该通过下列方程来计算:
$$\text{排序结束时共执行的最大次数} = (n) * (n-1) / 2 / \textcircled{S2} [\text{执行次数}]$$

例如, 如果 $n=10$ 且 $\textcircled{S2}=1$, 则完成排序将需要 $10 * (10-1) / 2 / 1 = 45$ 个扫描周期。
如果上例中的 $\textcircled{S2}$ 等于 2, 则排序结束时的扫描周期数将为 $45 / 2 = 22.5$, 小数部分四舍五入而变为 23。
- (5) $\textcircled{S3}$ (完成软元件) 指定的软元件在 SORT 指令执行前为 OFF, 而在排序结束时为 ON。
因为在排序结束后, $\textcircled{S3}$ 指定的软元件保持为 ON 状态, 所以如果需要, 用户必须将它变为 OFF。
- (6) 在 SORT 指令的执行期间, 从 $\textcircled{S3}$ 指定的软元件开始的 2 点为系统所使用。
因此, 用户不应该使用从 $\textcircled{S3}$ 指定的软元件开始的 2 点。
- (7) 如果在 SORT 指令的执行期间更改了 n 的值, 则排序运算将按照更改后的排序数据块数来执行。
- (8) 如果在执行排序运算的过程中命令变为 OFF, 则排序运算将中止。
当命令重新为 ON 时, 排序将从头开始。

DSORT

- (1) 对从 $\textcircled{S1}$ 开始的 n 点 BIN 32 位数据按升序或者降序进行排序 (重新排列数据)。
- (2) 排序类型由 SM703 的 ON 或 OFF 状态指定:
- 当 SM703 的状态为 OFF 时: 升序排序
 - 当 SM703 的状态为 ON 时: 降序排序



(3) 用 DSORT 指令执行排序需要若干个扫描周期。

在排序结束时共执行的最大次数除以执行一次排序运算所比较的数据块数(由 $\textcircled{S2}$ 指定)，所得到的结果值就是到排序结束共执行的扫描周期数。(小数部分四舍五入)。

如果增大 $\textcircled{S2}$ 的值，则在排序结束时扫描周期数减少，但每次扫描的时间变长。

(4) 在排序结束时共执行的最大次数应该通过下列方程来计算：

$$\text{排序结束时共执行的最大次数} = (n) * (n-1) / 2 / \textcircled{S2} [\text{执行次数}]$$

例如，如果 $n=10$ 且 $\textcircled{S2}=1$ ，则完成排序将需要 $10 * (10-1) / 2 / 1 = 45$ 个扫描周期。

如果上例中的 $\textcircled{S2}$ 等于 2，则排序结束时的扫描周期数将为 $45 / 2 = 22.5$ ，小数部分四舍五入而变为 23。

(5) $\textcircled{S1}$ (完成软元件)指定的软元件在 SORT 指令执行前为 OFF，而在排序结束时为 ON。

因为在排序结束后， $\textcircled{S1}$ 指定的软元件保持为 ON 状态，所以如果需要，用户必须将它变为 OFF。

(6) 在 DSORT 指令的执行期间，从 $\textcircled{S2}$ 指定的软元件开始的 2 点为系统所使用。

因此，用户不应该使用从 $\textcircled{S2}$ 指定的软元件开始的 2 点。

(7) 如果在 DSORT 指令的执行期间更改了 n 的值，则排序运算将按照更改后的排序数据块数来执行。

(8) 如果在执行排序运算的过程中将命令为 OFF，则排序运算将中止。

当命令重新为 ON 时，排序将从头开始。

[运行错误]

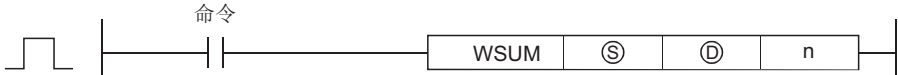
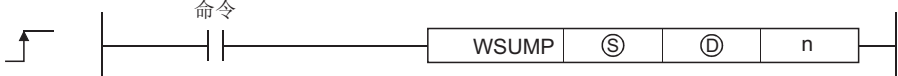
(1) 在以下发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。

- 对 SORT(P)指令而言，从 $\textcircled{S1}$ 指定的软元件开始的 n 点范围超出了相应软元件范围。(错误代码:4101)
- 对 DSORT(P)指令而言，从 $\textcircled{S1}$ 指定的软元件开始的 $2 * n$ 点范围超出了相应软元件范围。(错误代码:4101)
- $\textcircled{S2}$ 为 0 或负的值。(错误代码:4100)
- $\textcircled{S2}+0$ 的值大于 n 的值。(错误代码:4100)
- $\textcircled{S2}+1$ 的值大于 $\textcircled{S2}+0$ 的值。(错误代码:4100)

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.5.13 16 位数据的总和计算 (WSUM, 与 WSUMP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [J][K][L]		特殊功能模 块 [U][V][G][L]	变址寄存器 Zn
	位	字		位	字		常量 K, H
⑤	—	○		—			—
⑥	○	○		○			—
n	○	○		○			○

[指令符号]	[执行条件]
WSUM	
WSUMP	

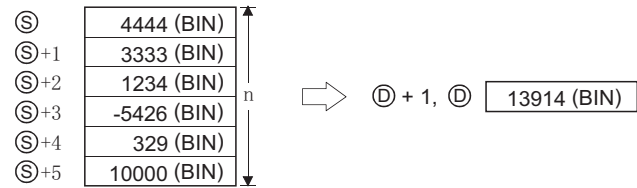
[设定数据]

设定数据	含义	数据类型
⑤	存储参与总和计算的数据的软元件的起始号	BIN 16 位
⑥	存储总和值的软元件的起始号	BIN 32 位
n	数据块数	BIN 16 位

[功能]

WSUM

(1) 将从⑤指定的软元件开始的 n 块 16 位 BIN 数据全都加起来, 并把结果存储在⑥指定的软元件中。



[运行错误]

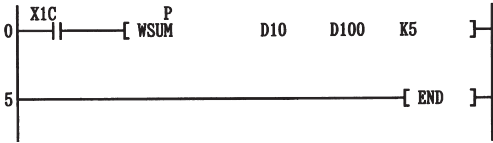
- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 从⑤软元件开始的 n 点范围超出了相关软元件。(错误代码:4101)

[程序示例]

WSUM

(1) 当 X1C 变为 ON 时，以下程序将 D10 到 D14 中的 16 位 BIN 数据加起来，并存储在 D100 和 D101 中。

[梯形图模式]



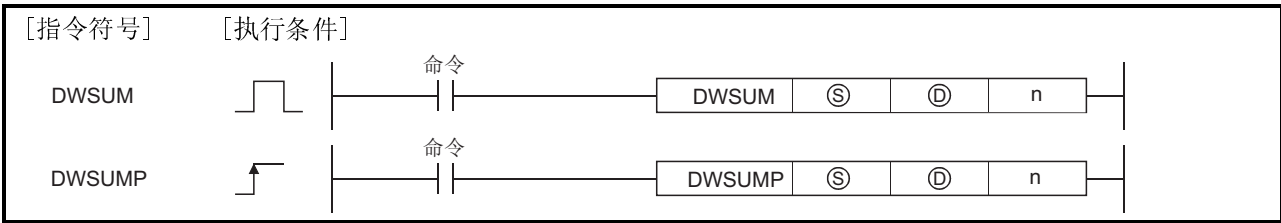
[列表模式]

步	指令	软元件
0	LD	X1C
1	WSUMP	D10 D100 K5
5	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.5.14 32 位数据总和计算(DWSUM, DWSUMP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 [J[]\[]]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		常量 K, H
Ⓢ	—	○			—		—
Ⓓ	○	○			—		—
n	○	○			○		○



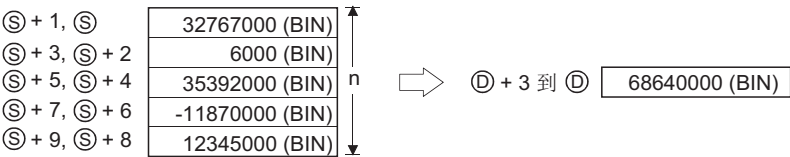
[设定数据]

设定数据	含义	数据类型
Ⓢ	存储用于总和计算的数据的软元件的起始软元件号	BIN 32 位
Ⓓ	存储总和计算结果的软元件的起始软元件号	BIN 64 位
n	数据块的数目	BIN 16 位

[功能]

DWSUM

(1) 从Ⓢ指定的软元件开始的全部 4 点（4 字）32 位 BIN 数据相加，并将结果存储在Ⓓ指定的软元件上。

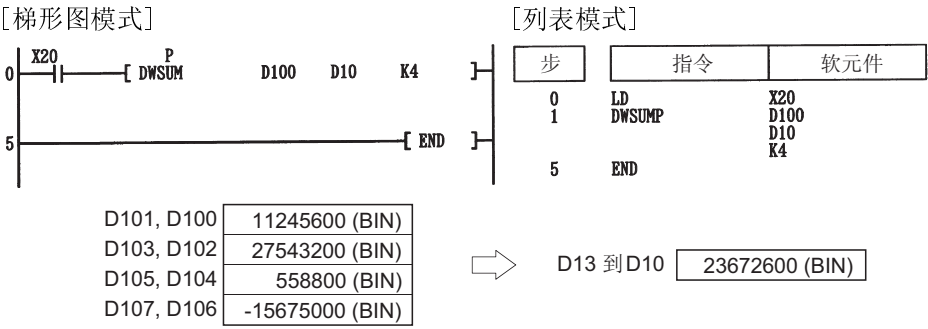


[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 从⑤软元件开始的 n 点的范围超过了相关软元件。(错误代码：4101)

[程序示例]

- (1) 当 X20 为 ON 时，以下程序将 D100 到 D107 的 32 位 BIN 数据相加，并将结果存储在 D10 到 D13 上。

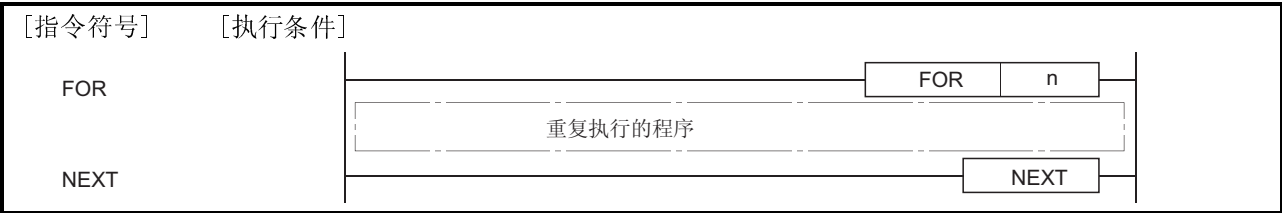


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.6 结构化程序指令

7.6.1 FOR 到 NEXT 指令循环 (FOR, NEXT)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][][]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
n	○								—

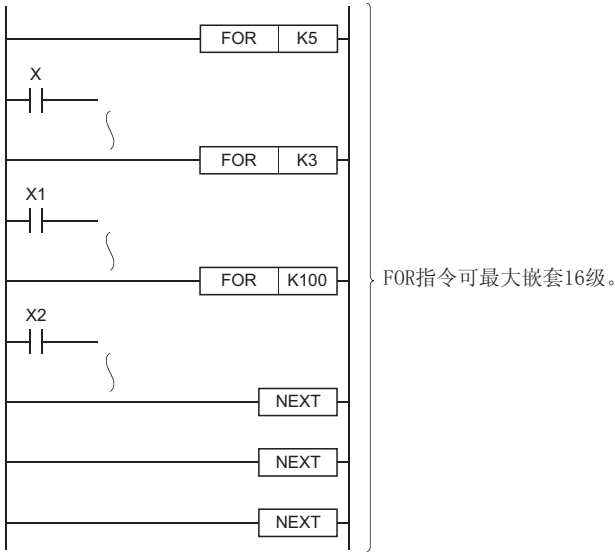


[设定数据]

设定数据	含义	数据类型
n	FOR 到 NEXT 循环的重复次数 (1 到 32767)	BIN 16 位

[功能]

- (1) 当 FOR 到 NEXT 循环中的处理无条件执行了 n 次时，将执行跟着 NEXT 指令之后的步。
- (2) n 的值可指定在 1 到 32767 之间。
如果指定为 -32768 到 0 之间的一个值，则当作 n=1 执行。
- (3) 如果不想执行在 FOR 到 NEXT 循环内调用的处理，则使用 CJ 或 SCJ 指令跳过去。
- (4) FOR 指令可最大嵌套 16 级。



[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 在一个 FOR 指令执行之后，但在一个 NEXT 指令执行之前，
执行了一个 END(FEND) 指令。
 - 在一个 FOR 指令执行之前，执行了一个 NEXT 指令。
 - 已经在 FOR 到 NEXT 循环内插入一个 STOP 指令。
 - 当已经嵌套使用 FOR 指令时，遇到第 17 个 FOR 指令。

(错误代码：4200)

(错误代码：4201)

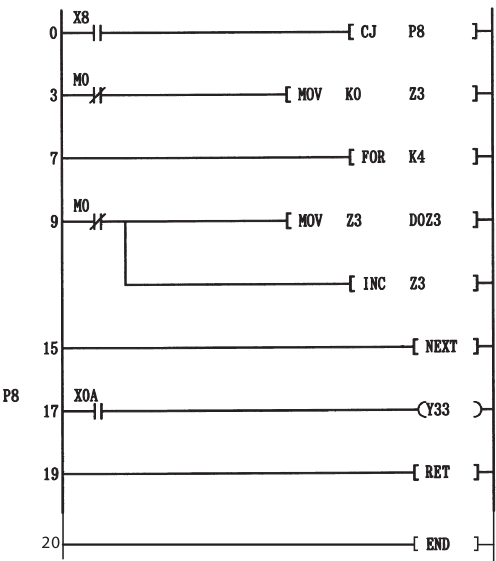
(错误代码：4200)

(错误代码：4202)

[程序示例]

(1) 当 X8 为 OFF 时，以下程序执行 FOR 到 NEXT 循环，当 X8 为 ON 时，不执行。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X8
1	CJ	P8
3	LDI	M0
4	MOV	K0
		Z3
7	FOR	K4
9	LDI	M0
10	MOV	Z3
		D0Z3
13	INC	Z3
15	NEXT	
16		P8
17	LD	X0A
18	OUT	Y33
19	RET	
20	END	

备注

- 1) 为了在 FOR 到 NEXT 循环执行期间强制结束该循环的重复执行，请插入一个 BREAK 指令。
有关 BREAK 指令使用的详细内容请参见 7.6.2 节。
- 2) 在 FOR 和 NEXT 之间使用 EGP/EGF 指令执行一个变址修改程序的脉冲操作。
EGP/EGF 指令的详情请参见 5.2.5 节。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.6.2 FOR 到 NEXT 指令循环的强制结束(BREAK, BREAKP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]K[]		特殊功能模 块 U[]V[]G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				P
①	○						—	—	
Pn	—						—	○	

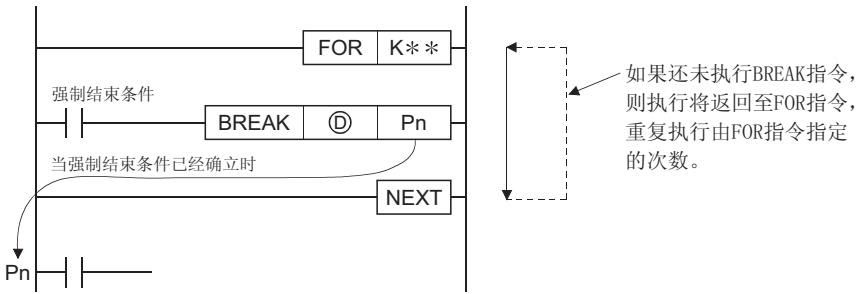
[指令符号]	[执行条件]
BREAK	
BREAKP	

[设定数据]

设定数据	含义	数据类型
①	存储剩余重复次数的软元件的号码	BIN 16 位
Pn	当重复处理结束时的分支目标指针号	软元件名称(指针)

[功能]

(1) 强制结束 FOR 到 NEXT 指令循环的重复处理，切换到 Pn 指定的指针指向的处理操作。



- (2) 将在 FOR 到 NEXT 循环被强制结束时剩余的重复次数存储到①上。
但是，执行 BREAK 指令时的循环也包括在剩余的重复次数中。
- (3) 仅在 FOR 到 NEXT 指令循环执行期间才可使用 BREAK 指令。
- (4) 仅当只有一级嵌套时才可使用 BREAK 指令。
如果当有多重嵌套时强制结束，则执行同嵌套级数相同数量的 BREAK 指令。

[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 在一个非 FOR 到 NEXT 的指令循环内使用 BREAK 指令。

(错误代码: 4203)

● Pn 指定的指针的跳转目的地不存在。

(错误代码: 4210)

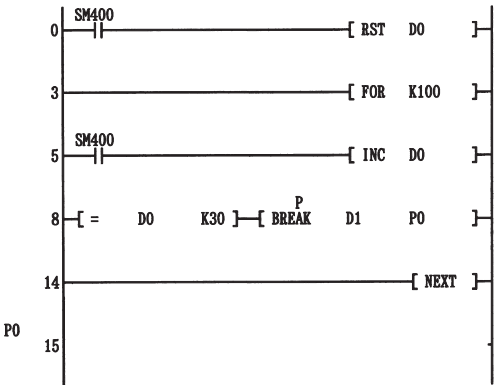
● 另一个程序文件的指针被指定给 Pn。

(错误代码: 4210)

[程序示例]

- (1) 当 D0 值到达 30 时（当 FOR 到 NEXT 循环被执行 30 次时），以下程序强制结束 FOR 到 NEXT 循环。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	RST	D0
3	FOR	K100
5	LD	SM400
6	INC	D0
8	LD=	D0
		K30
11	BREAKP	D1
		P0
14	NEXT	
15		P0

备注

※:当执行 BREAK 指令时，将值 71 存储在 D1 上。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.6.3 子程序调用(CALL, CALLP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[N]C		特殊功能模 块 U[N]G	变址寄存器 Zn	常量	其它 P
	位	字		位	字				
P	—	—	—					○	
① 到⑤ (除 F 之外)	○	○	○					—	

[指令符号]	[执行条件]
CALL	
CALLP	
CALL	
CALLP	

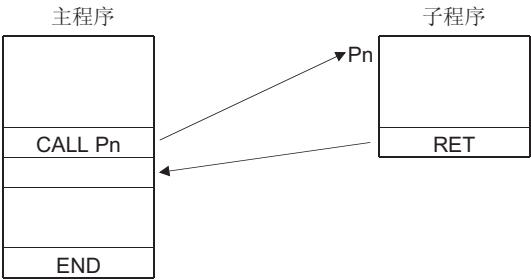
[设定数据]

设定数据	含义	数据类型
Pn	子程序的起始指针号	软元件名称
①到⑤	作为变量传送到子程序的软元件号	位 BIN 16 位 BIN 32 位

[功能]

(1) 当执行 CALL(P) 指令时，执行由 Pn 指定的程序的子程序。

CALL(P) 指令可以执行由同一个程序文件中的一个指针指定的子程序和由一个公共指针指定的子程序。



(2) 当功能软元件 (FX, FY, FD) 被一个子程序使用时, 用与功能软元件对应的⑨1到⑨5指定软元件。
由⑨1到⑨5指定的软元件的内容简要说明如下。

(a) 在子程序执行之前, 将位数据传送到 FX, 将字数据传送到 FD。

(b) 在子程序执行之后, 将 FY 和 FD 的内容传送到相应的软元件。

(c) 用于功能软元件的处理单元如下:

- FX, FY : 位
- FD : 4 字单元

即将处理的数据大小依据变量中指定的软元件会有所不同。

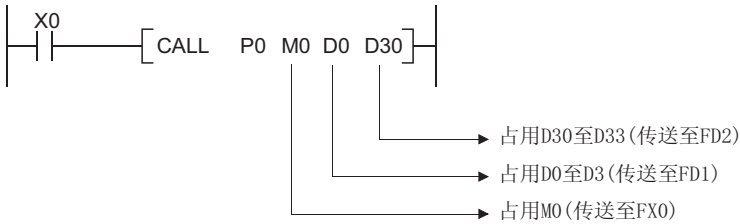
指定为功能软元件的软元件应该确保满足该数据量。

如果未能确保该数据量, 将发生错误。

功能软元件	使用软元件	数据大小	备注
● FX	● 位软元件	1 点	
● FY	● 当为字软元件进行位指定时	1 位	
● FD	● 当使用位软元件的位数指定时*	4 字	
	● 字软元件	4 字	数据大小根据 将要使用的指令有所不同。

*: 当⑨1到⑨5指定的软元件号在位软元件的位数指定处不是 16 的倍数时, 不会发生错误。

[主程序]



(3) ⑨1到⑨5可以和 CALL (P) 指令一起使用。

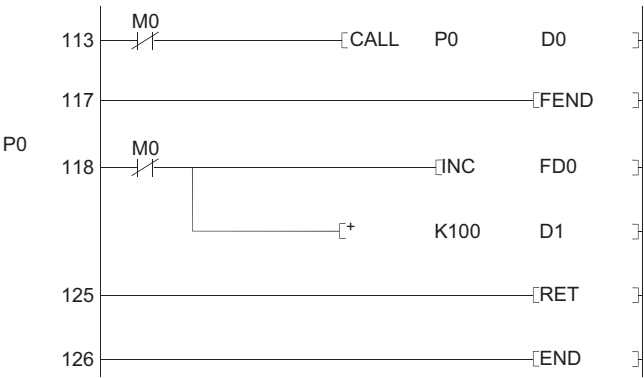
(4) 子程序将要使用的功能软元件数必须同 CALL (P) 指令中的变量数相同。
而且, 功能软元件和使用的 CALL (P) 变量类型应该完全一致。

(5) 由 CALL (P) 指令指定的软元件号不应重叠。
如果它们重叠, 将无法获得准确的计算。

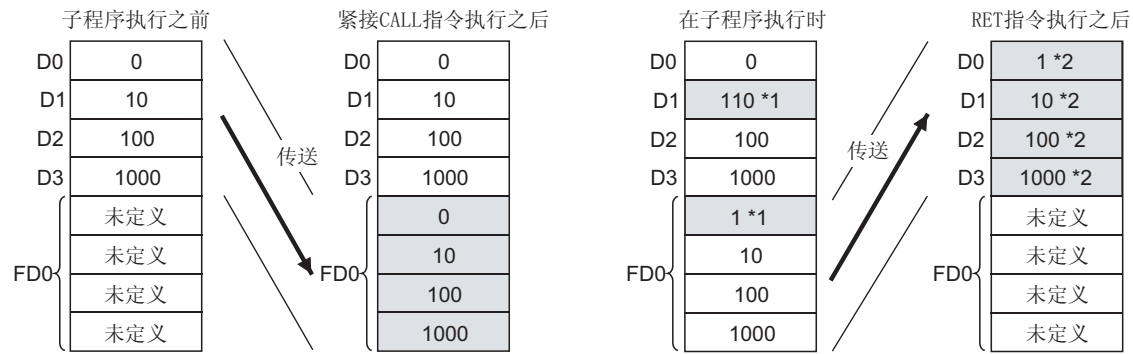
(6) CALL 指令变量中使用的软元件不应在子程序中使用。
如果使用，将无法获得准确计算。（参见以下程序示例。）

以下程序示例显示当在子程序中将 D0 指定为 FD0 且在子程序中使用 D1 时进行的操作。

[程序示例]



[子程序执行之后进行的运算]



*1: 存储子程序的执行结果。
*2: 被功能软元件的值替换。
D1 不反映子程序中的运算结果。

(7) CALL (P) 指令最多可以有 16 级嵌套。
然而，该 16 级是 CALL (P)、FCALL (P)、ECALL (P)、EFCALL (P) 和 XCALL 指令中嵌套级数的总和。

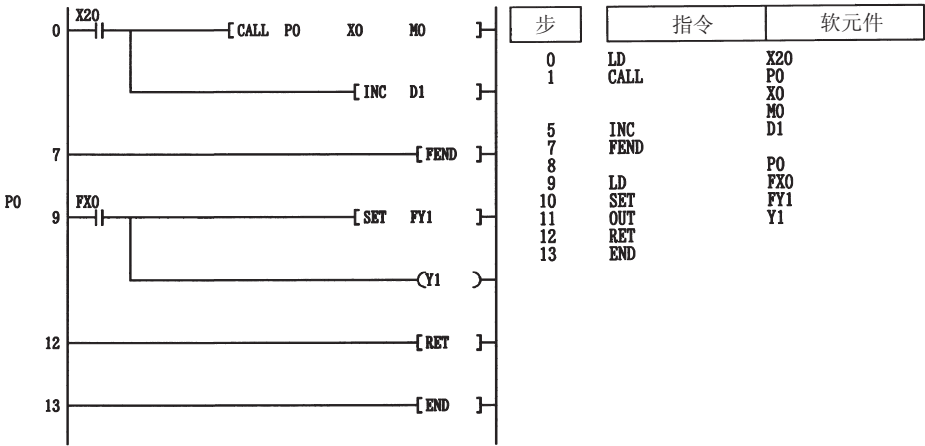
- (8) 子程序内变为 ON 的软元件将被闭锁，即使子程序未执行。
子程序执行期间变为 ON 的软元件可以通过执行 FCALL (P) 指令被变为 OFF。

[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 为变量指定的软元件无法确保数据大小。 (错误代码: 4101)
 - 在 CALL (P) 指令执行之后，在执行 RET 指令之前执行了 END、FEND、GOEND 或者 STOP 指令。 (错误代码: 4211)
 - 在执行 CALL (P) 指令之前执行 RET 指令。 (错误代码: 4212)
 - 执行第 17 级嵌套。 (错误代码: 4213)
 - CALL (P) 指令指定的指针没有子程序。 (错误代码: 4210)

[程序示例]

- (1) 当 X20 变为 ON 时，以下程序执行一个带有变量的子程序。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

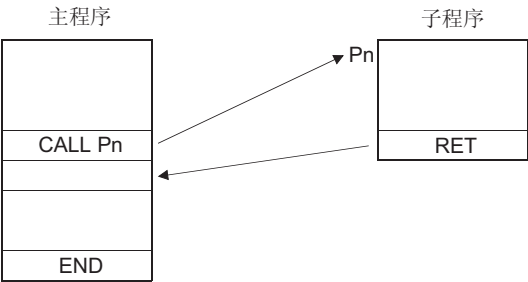
7.6.4 从子程序返回(RET)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[] []		特殊功能模 块 U[] N[] G[]	变址寄存器 Zn	常量	其它
	位	字		位	字				
—	—								

[指令符号]	[执行条件]
RET	<div><div>RET</div></div>

[功能]

- (1) 表示子程序结束
- (2) 当执行 RET 指令时, 返回到调用子程序的 CALL (P)、FCALL (P)、ECALL (P)、EFCALL (P) 或者 XCALL 指令之后的步。



[运行错误]

- (1) 在以下发生运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SDO 中。
- 在 CALL (P)、FCALL (P)、ECALL (P)、EFCALL (P) 或者 XCALL 指令执行之后, 在执行 RET 指令之前执行 END、FEND、GOEND 或者 STOP 指令。

(错误代码: 4211)
 - 在执行 CALL (P)、FCALL (P)、ECALL (P)、EFCALL (P) 或者 XCALL 指令之前执行 RET 指令。

(错误代码: 4212)

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.6.5 子程序输出 OFF 调用 (FCALL, FCALLP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [J][N][C]		特殊功能模 块 [I][N][C]	变址寄存器 Zn
	位	字		位	字		常量
P	—	—				—	○
①到⑤ (除 F 之外)	○	○				○	—

[指令符号]	[执行条件]
FCALL	
FCALLP	
FCALL	
FCALLP	

[设定数据]

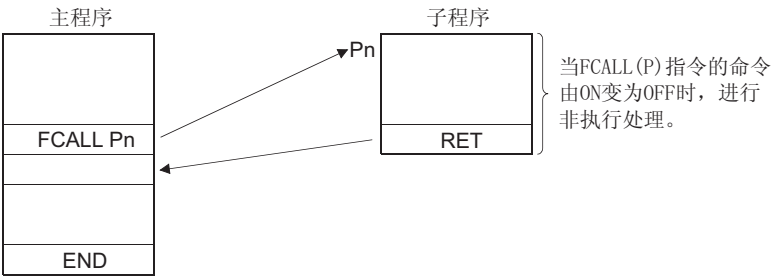
设定数据	含义	数据类型
Pn	子程序的起始指针号	软元件名称
①到⑤	作为变量传到子程序的软元件号	位 BIN 16 位 BIN 32 位

[功能]

(1) 当执行 FCALL (P) 时，执行由 Pn 指定的指针的子程序的非执行处理。

FCALL (P) 指令可以执行由相同程序文件中一个指针指定的子程序和由一个公共指针指定的子程序。

(a) 非执行处理同单线圈指令的条件触点为 OFF 时进行的处理完全相同。



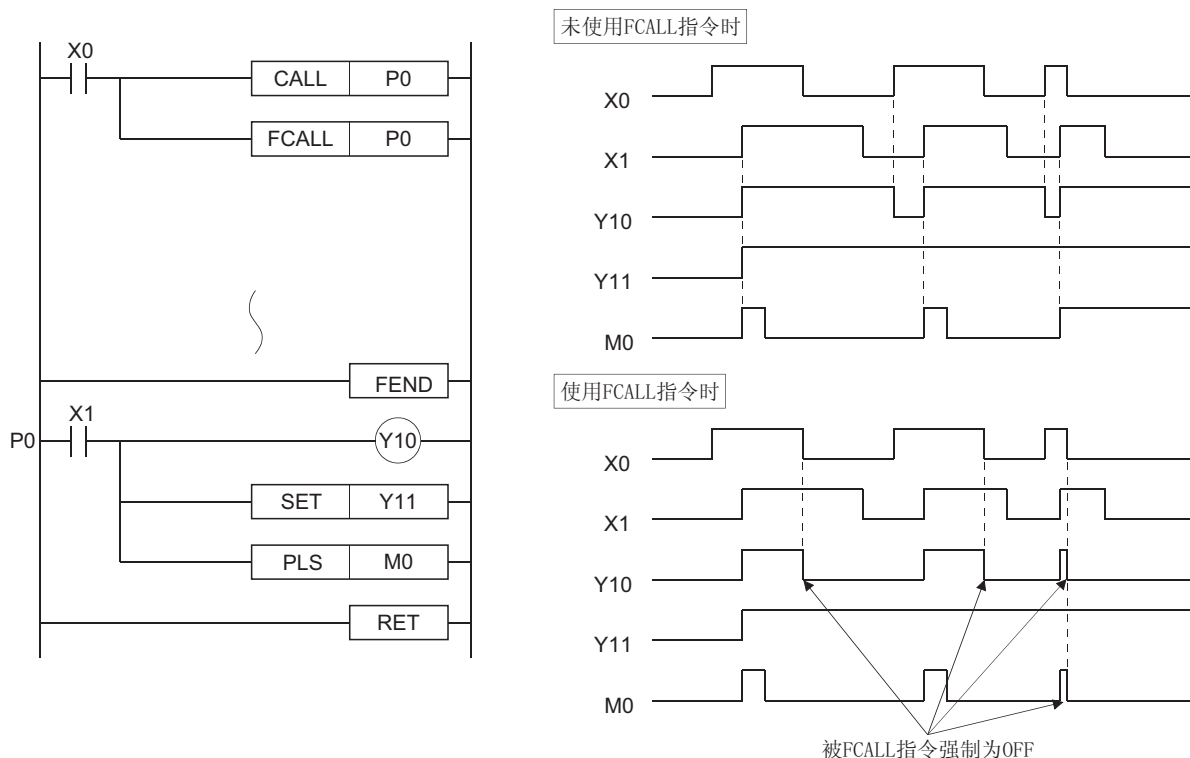
(b) 不管单个触点的状态是 ON 还是 OFF，非执行处理之后的单线圈指令的运算结果如下：

- | | |
|----------------------------|------------------------|
| • OUT指令..... | 强制OFF |
| • SET指令 | } 保持状态 |
| • RST指令 | |
| • SFT指令 | |
| • 基本指令 | |
| • 应用指令 | } 等同于条件触点为OFF时的处理 |
| • PLS指令 | |
| • 脉冲生成指令
([] P) | |
| • 低速定时器和高速定时器的
当前值..... | 0 |
| • 累计定时器的当前值 | } 保持 |
| • 计数器的当前值 | |

(2) FCALL (P) 指令和 CALL (P) 指令结合在一起使用。

(3) 如果未执行 FCALL (P) 指令（仅使用 CALL (P) 指令的情况），则当子程序指令变 OFF 时，由于还未执行子程序，所以将保存每个单线圈指令的输出状态。

当执行 FCALL (P) 指令时，OUT 指令和 PLS 指令（包括 [] P 指令）可能被强制变为 OFF，以进行子程序非执行处理。



(4) 当功能软元件 (FX, FY, FD) 被一个子程序使用时, 通过⑤1到⑤5指定与功能软元件对应的软元件。
由⑤1到⑤5指定的软元件的内容简要说明如下。

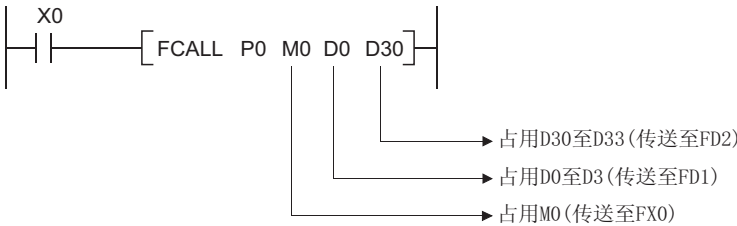
- (a) 在子程序执行之前, 将位数据传送到 FX, 将字数据传送到 FD。
- (b) 在子程序执行之后, 将 FY 和 FD 的内容传送到相应的软元件。
- (c) 功能软元件处理单元如下:

- FX, FY : 位
 - FD : 4 字单元
- 即将处理的数据大小依据变量中指定的软元件会有所不同。
指定为功能软元件的软元件应该确保满足该数据量。
如果未能确保该数据量, 将发生错误。

功能软元件	使用软元件	数据大小	备注
● FX	● 位软元件	1 点	
● FY	● 当为字软元件进行位指定时	1 位	
● FD	● 当使用位软元件的位数指定时*	4 字	FD 的高 2 字变成 0
	● 字软元件	4 字	

*: 当⑤1到⑤5指定的软元件号在位软元件位数指定处不是 16 的倍数时, 不会发生错误。

[主程序]



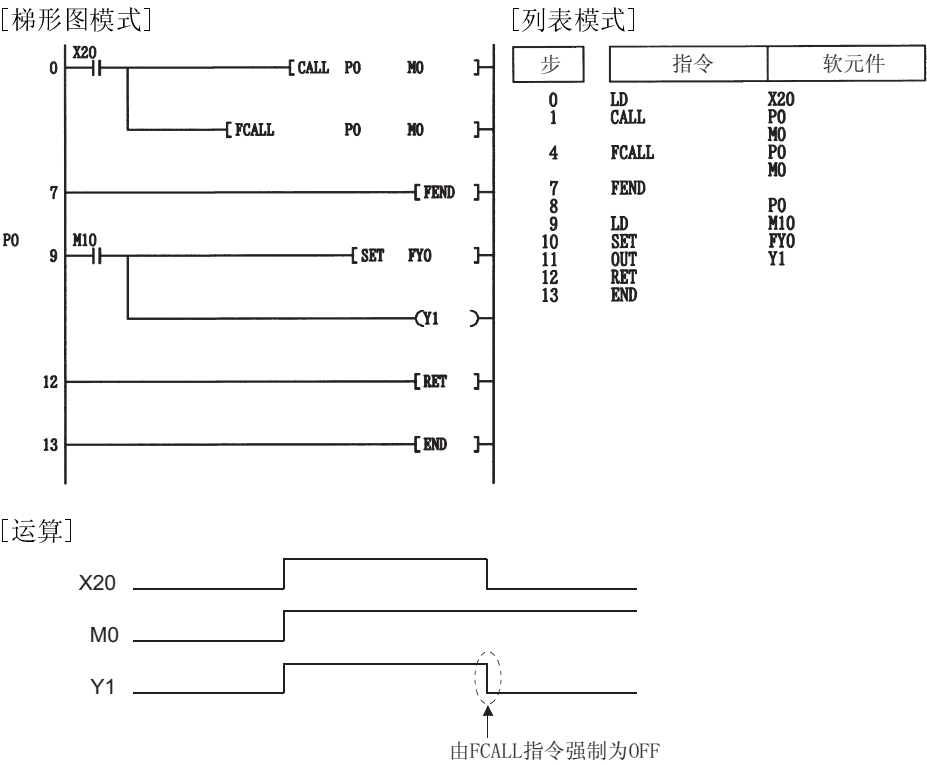
- (5) FCALL (P) 指令可以使用⑤1到⑤5。
- (6) FCALL (P) 指令最多可以有 16 级嵌套。
然而, 该 16 级是 CALL (P)、FCALL (P)、ECALL (P)、EFCALL (P) 和 XCALL 指令中嵌套级数的总和。

[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 为变量指定的软元件无法确保数据大小。 (错误代码: 4101)
 - 在 FCALL (P) 指令执行之后，在执行 RET 指令之前执行 END、FEND、GOEND 或者 STOP 指令。 (错误代码: 4211)
 - 在执行 FCALL (P) 指令之前执行 RET 指令。 (错误代码: 4212)
 - 执行第 17 级嵌套。 (错误代码: 4213)
 - FCALL (P) 指令指定的指针的子程序不存在。 (错误代码: 4210)

[程序示例]

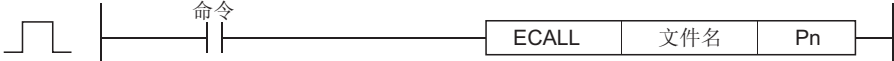
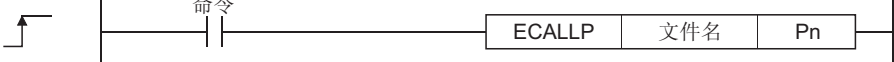


- (1) 当 X20 为 ON 时，以下程序执行一个带有变量的子程序，当 X20 从 ON 变到 OFF 时，强制进行非执行处理。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.6.6 程序文件之间的子程序调用 (ECALL, ECALLP)

设定 数据	可用软元件									
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J K L		特殊功能模 块 U V W G	变址寄存器 Zn	常量		其它 P
	位	字		位	字			K, H	\$	
文件名	—	○		—				○	—	
P	—	—		—				—	○	
④ 到⑤ (除 F 之外)	○	○		○				—	—	

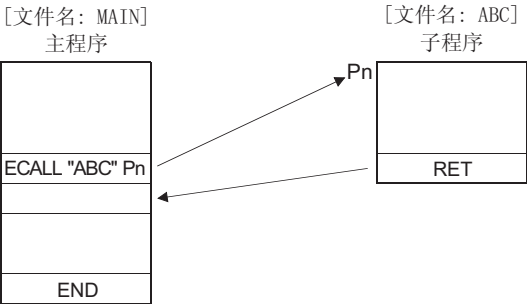
[指令符号]	[执行条件]
ECALL	
ECALLP	
ECALL	
ECALLP	

[设定数据]

设定数据	含义	数据类型
文件名	即将调用的程序文件的名称	字符串
Pn	子程序的起始指针号	软元件名称
④ 到 ⑤	作为变量传到子程序的软元件号	位 BIN 16 位 BIN 32 位

[功能]

- (1) 当执行 ECALL (P) 指令时，以指定的程序文件名执行 Pn 指定的指针的子程序。
可使用 ECALL (P) 指令从不同的程序文件中调用使用局部指针的子程序。

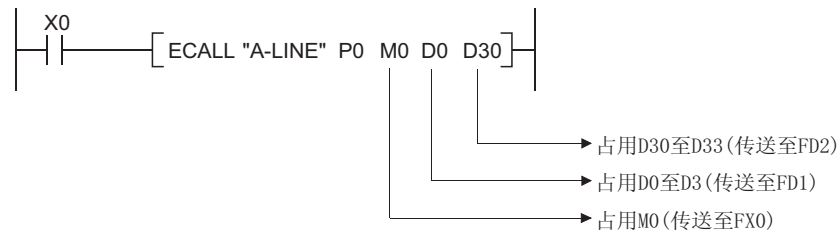


- (2) 文件名仅可以指定存储在内存/程序内存（驱动器 0）中的程序文件的名称。
- (3) 指定文件名时，不需要扩展名。
- (4) 当功能软元件 (FX, FY, FD) 被一个子程序使用时，通过⑨到㉑指定与功能软元件对应的软元件。
- 由⑨到㉑指定的软元件的内容简要说明如下。
- (a) 在子程序执行之前，将位数据传送到 FX，将字数据传送到 FD。
- (b) 在子程序执行之后，将 FY 和 FD 的内容传送到相应的软元件。
- (c) 功能软元件的处理单元如下：
- FX, FY ： 位
 - FD ： 4 字单元
- 即将处理的数据大小依据变量中指定的软元件会有所不同。
- 指定为功能软元件的软元件应该确保该数据量。
- 如果未能确保该数据量，将发生错误。

功能软元件	使用软元件	数据大小	备注
● FX	● 位软元件	1 点	数据大小根据将要使用的指令有所不同。
● FY	● 当为字软元件进行位指定时	1 位	
● FD	● 当使用位软元件的位数指定时*	2 字	
	● 字软元件	4 字	

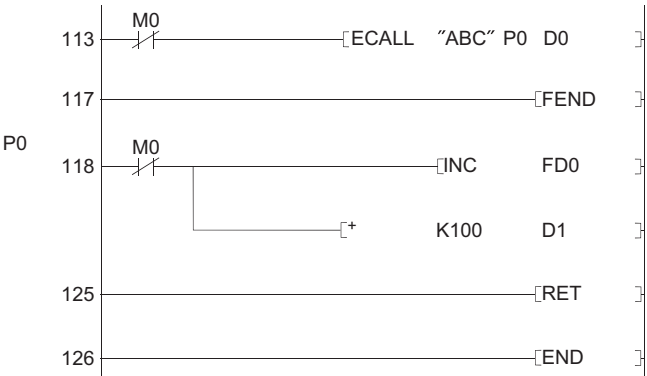
*:当⑨到㉑指定的软元件号在位软元件位数指定不是 16 的倍数时，不会发生错误。

[主程序]

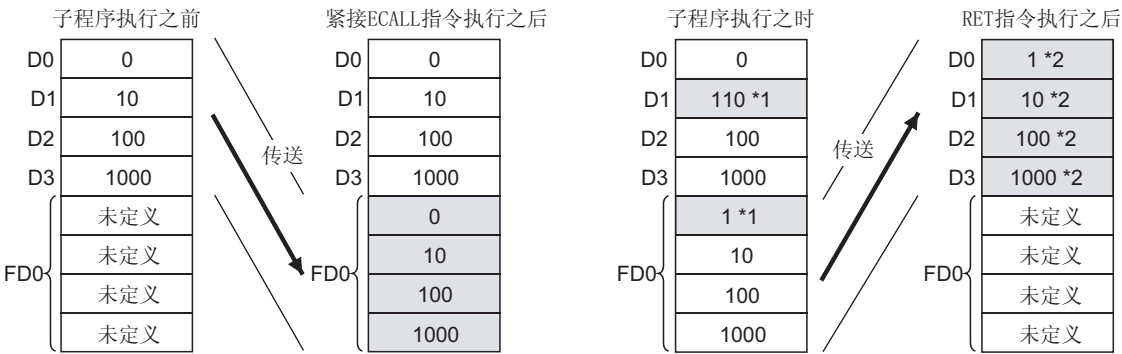


- (5) ⑨ 到 ㉑ 可以通过 ECALL 指令使用。
- (6) ECALL 指令变量中使用的软元件不能应用于子程序。
- 如果使用，则无法获得准确的计算。（参见以下程序示例。）
- 以下程序示例显示当在子程序中将 D0 指定为 FD0 且在子程序中使用 D1 时进行的运算。

[程序示例]



[子程序执行之后进行的运算]



*1: 存储子程序的执行结果。
*2: 由功能软元件值替换。
D1 不反映子程序中的运算结果。

- (7) 在 ECALL (P) 指令中由变量指定的软元件号不应重叠。
如果重叠，则无法获得准确的计算。
- (8) ECALL (P) 指令最多可以有 16 级嵌套。
然而，该 16 级是 CALL (P)、FCALL (P)、ECALL (P) 和 EFCALL (P) 指令中嵌套的总级数。
- (9) 子程序内变为 ON 的软元件将被锁住，即使子程序未执行。
子程序执行期间变为 ON 的软元件可以通过 FCALL (P) 指令被变为 OFF。

[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 为变量指定的软元件无法确保数据大小。

(错误代码: 4101)
- 在 FCALL (P) 指令执行之后，在执行 RET 指令之前执行 END、FEND、GOEND 或者 STOP 指令。

(错误代码: 4211)
- 在执行 RET 指令之前，且在执行 ECALL (P) 指令之后，执行 END、FEND、GOEND、STOP 指令。

(错误代码: 4211)
- 在执行 ECALL (P) 指令之前执行 RET 指令。

(错误代码: 4212)
- 执行第 17 级嵌套。

(错误代码: 4213)
- ECALL (P) 指令指定指针的子程序不存在。

(错误代码: 4210)
- 指定文件不存在。

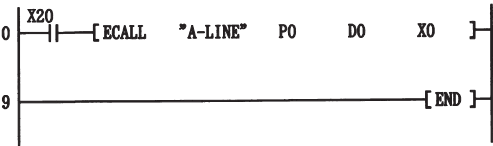
(错误代码: 4210)
- 指定文件无法执行。

(错误代码: 2411)

[程序示例]

(1) 当 X20 为 ON 时，以下程序执行 P0 A-LINE.QPG。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X20
1	ECALL	"A-LINE"
		P0
		D0
		XO
9	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.6.7 程序文件之间的子程序输出 OFF 调用 (EFCALL, EFCALLP)

设定 数据	可用软元件									
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常量		其它 P
	位	字		位	字			K, H	\$	
文件名	—	○		—				○	—	
P	—	—		—				—	○	
⑤① 到 ⑤⑤ (除 F 之外)	○	○		○				—	—	

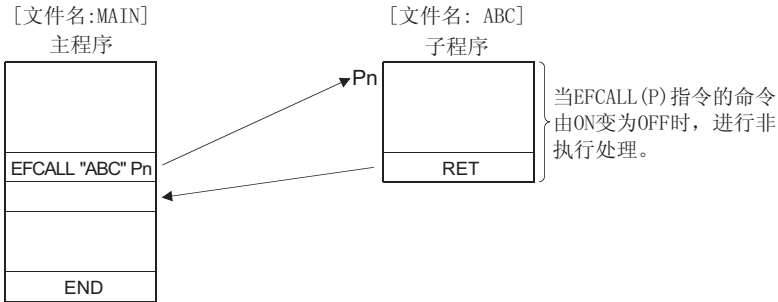
[指令符号]	[执行条件]
EFCALL	
EFCALLP	
EFCALL	
EFCALLP	

[设定数据]

设定数据	含义	数据类型
文件名	即将调用的程序文件的名称	字符串
Pn	子程序的起始指针号	软元件名称
⑤①到⑤⑤	作为变量传到子程序的软元件号	位 BIN 16 位 BIN 32 位

[功能]

- (1) 当执行 EFCALL (P) 指令时, 执行 Pn 指定的指针的子程序的非执行处理。
- 〔 也可使用EFCALL (P) 指令从不同程序文件中调用使用局部指针的子程序。 〕
- (a) 非执行处理同单线圈指令的状态触点为 OFF 时进行的处理完全相同。

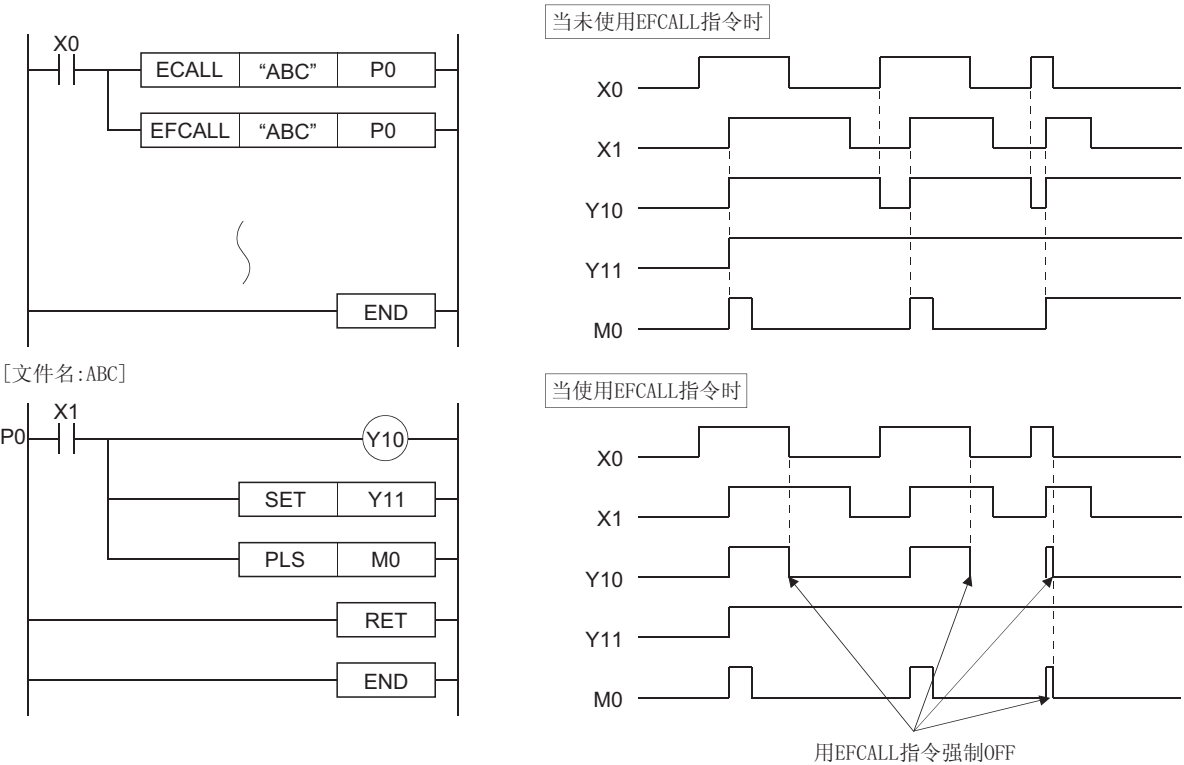


(b) 不管状态触点的 ON/OFF 状态如何，非执行处理之后的单线圈指令的运算结果如下：

- OUT指令.....强制为OFF
 - SET指令
 - RST指令
 - SFT指令
 - 基本指令
 - 应用指令
 - PLS指令
 - 脉冲生成指令
([] P)
 - 低速定时器和高速定时器的
当前值.....0
 - 累计定时器当前值
 - 计数器当前值
- 保持状态
- 等同于条件触点为OFF时的
处理
- 保持

(2) EFCALL (P) 指令与 ECALL (P) 指令结合在一起使用。

(3) 如果未执行 EFCALL (P) 指令（仅使用 ECALL (P) 指令的情况），则当子程序命令变 OFF 时，由于还未执行子程序，所以将保存每个单线圈指令的输出状态。
当执行 EFCALL (P) 指令时，对子程序进行非执行处理，因此可以将 OUT 指令和 PLS 指令（包括 [] P 指令）强制变为 OFF。



(4) 仅可指定存储在内置内存/程序内存（驱动器 0）中的程序文件作为文件名。

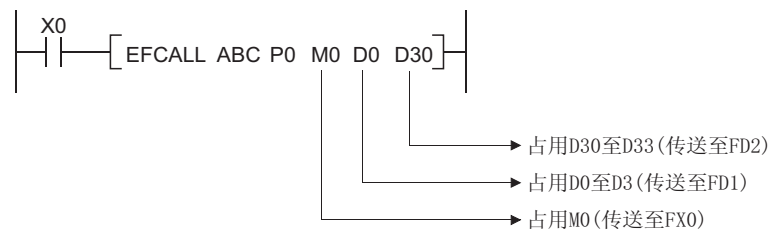
(5) 指定的文件名不需要扩展名。

- (6) 当功能软元件 (FX, FY, FD) 被一个子程序使用时, 通过⑤1到⑤5指定与功能软元件对应的软元件。
- (a) 在子程序执行之前, 将位数据传送至 FX, 将字数据传送至 FD。
- (b) 在子程序执行之后, 将 FY 和 FD 的内容传送至相应的软元件。
- (c) 功能软元件的处理单元如下:
- FX, FY : 位
 - FD : 4 字单元
- 即将处理的数据大小依据变量中指定的软元件会有所不同。
指定为功能软元件的软元件应该确保该数据量。
如果未能确保该数据量, 将发生错误。

功能软元件	使用软元件	数据大小	备注
● FX	● 位软元件	1 点	FD 的高 2 字变为 0
● FY	● 当为字软元件进行了位指定时	1 位	
● FD	● 当使用位软元件的位数指定时*	2 字	
	● 变址寄存器	4 字	

*: 当⑤1到⑤5指定的软元件号在位软元件的位数指定处不是 16 的倍数时, 不会发生错误。

[主程序]



- (7) 可通过 EFCALL (P) 指令使用⑤1到⑤5。
- (8) 由于子程序使用的功能软元件数必须同 EFCALL (P) 指令使用的变量数完全相同。
此外, 功能软元件应该同 EFCALL (P) 指令使用的变量类型相一致。
- (9) EFCALL (P) 指令最多可以用 16 级嵌套。
然而, 该 16 级是 CALL (P)、FCALL (P)、ECALL (P) 和 EFCALL (P) 指令中嵌套的总级数。

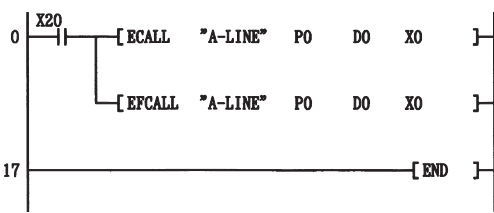
[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，如果还未进行变址修改，则将并且错误代码存储在 SD0 中。
- 为变量指定的软元件无法确保数据大小。(错误代码: 4101)
 - 在 EFCALL (P) 指令执行之后，在执行 RET 指令之前执行 END、FEND、GOEND 或者 STOP 指令。(错误代码: 4211)
 - 在执行 EFCALL (P) 指令之前执行 RET 指令。(错误代码: 4212)
 - 执行第 17 级嵌套。(错误代码: 4213)
 - EFCALL (P) 指令指定指针的子程序不存在。(错误代码: 4210)
 - 指定文件不存在。(错误代码: 4210)
 - 指定文件无法执行。(错误代码: 2411)

[程序示例]

- (1) 当 X20 为 ON 时，以下程序执行带有变量的子程序，当 X20 由 ON 变为 OFF 时，强制进行非执行处理。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X20
1	ECALL	"A-LINE" P0 D0 X0
9	EFCALL	"A-LINE" P0 D0 X0
17	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

＊1：系列号前五位数是04122或更大的。

7.6.8 子程序调用(XCALL)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][] G[]	变址寄存器 Zn	常量 K, H	其它 P
	位	字		位	字				
P	—	—	—						○
① 到 ⑤	○ (除 F 之外)	○	○						—

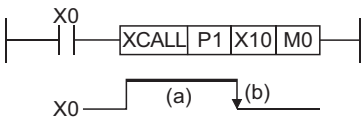


[设定数据]

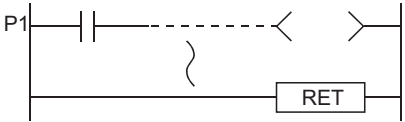
设定数据	含义	数据类型
Pn	● 子程序的起始指针号	软元件名称
①到⑤	● 作为变量传到子程序的软元件号	位 BIN 16 位 BIN 32 位

[功能]

(1) 当条件满足时，执行 Pn 指定的子程序，当状态从 ON 变为 OFF 时，进行非执行处理。

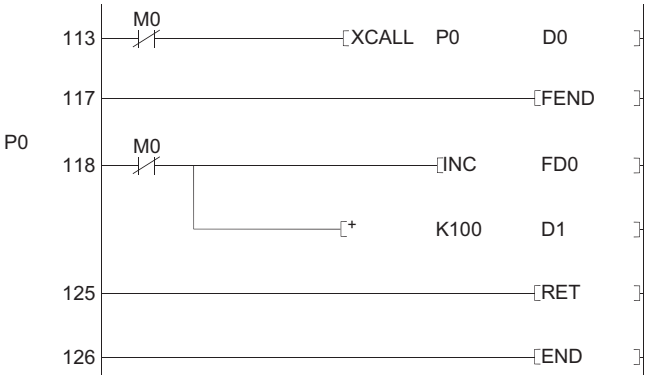


(a) 当 X0 为 ON 时，每次执行相应步骤时，在每个扫描周期内相应执行子函数程序 P1。

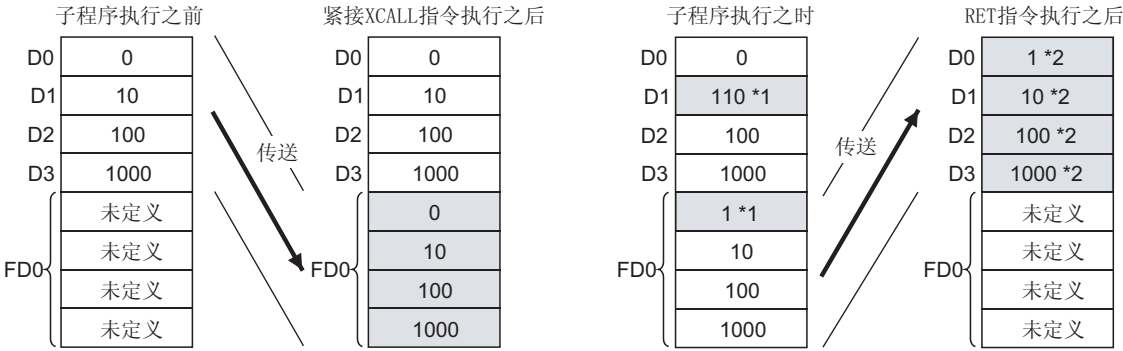


- (3) ⑤到⑤可由 XCALL 指令使用。
- (4) 由子程序使用的功能软元件数必须同 XCALL 指令使用的变量数完全相同。
而且，功能软元件和 XCALL 变量类型应该完全一致。
- (5) XCALL 指令变量所指定的软元件号不应重叠。
如果重叠，则无法获得准确计算。
- (6) XCALL 指令变量所使用的软元件不能应用于子程序。
如果使用，则无法获得准确计算。(请参见以下程序示例。)
- 以下程序示例显示当在子程序中将 D0 指定为 FD0 且在子程序中使用 D1 时进行的运算。

[程序示例]



[子程序执行之后进行的运算]



- *1: 存储子程序的执行结果。
- *2: 被功能软元件的值替换。
- D1 不反映子程序中的运算结果。

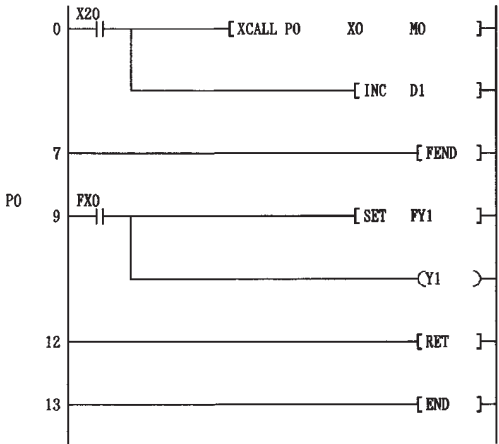
[运行错误]

- (1) 在以下情形下，发生运行错误，错误标志 (SM0) 变为 ON，错误代码存储在 SD0 中。
- 为变量指定的软元件无法确保数据大小。 (错误代码: 4101)
 - 在 XCALL (P) 指令执行之后，在执行 RET 指令之前执行 END、FEND、GOEND 或者 STOP 指令。 (错误代码: 4211)
 - 在执行 XCALL (P) 指令之前执行 RET 指令。 (错误代码: 4212)
 - 执行第 17 级嵌套。 (错误代码: 4213)
 - XCALL (P) 指令指定的指针没有子程序。 (错误代码: 4210)

[程序示例]

- (1) 当 X20 变为 ON 时，以下程序执行带有变量的子程序。

[梯形图模式]



[列表模式]

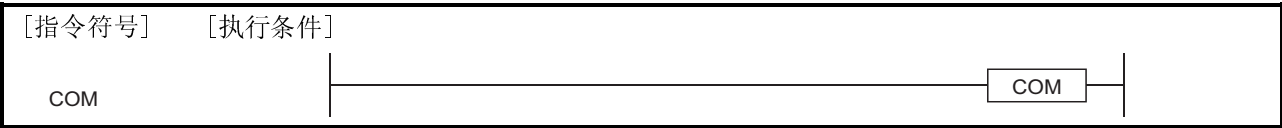
步	指令	软元件
0	LD	X20
1	XCALL	P0 X0 M0
5	INC	D1
7	FEND	
8		P0
9	LD	FX0
10	SET	FY1
11	OUT	Y1
12	RET	
13	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.6.9 刷新指令 (COM)

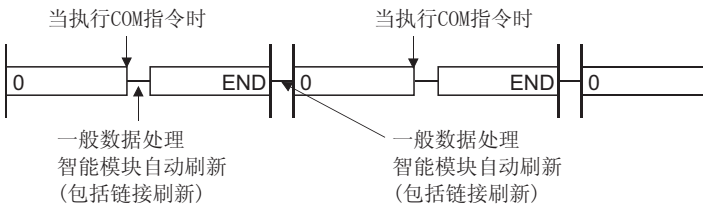
- (1) 适用于功能版本为 A 的基本型 QCPU、系列号为“04011”或更早的高性能型 QCPU 和过程控制 CPU
- 以下 CPU 型号的 COM 指令，请参见 9.11 节。
- 系列号为“04122”或之后的基本型 QCPU
 - 系列号为“04012”或之后的高性能型 QCPU

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块U[][G][]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
—	—								



[功能]

- (1) 在以下情况下，使用 COM 指令：
- (a) 要提高同远程 I/O 站的传送/接收处理速度时。
- (b) 要在执行数据链接期间，确保同使用不同扫描周期的其它站之间进行可靠的数据传送/接收。
- (2) 依据特殊继电器 SM775 是 ON 还是 OFF 状态，COM 指令的处理会有变化。
- 当 SM775 在 OFF 状态时：自动刷新，并进行一般的数据处理。*1, *2
 - 当 SM775 在 ON 状态时：仅进行一般数据处理。*1
- (3) 在即将执行 COM 指令的时候，CPU 模块将暂停顺控程序的处理，而进行与 END 处理期间智能功能模块的一般数据处理和自动刷新（包括链接刷新）相同的操作。
- 但是，不进行低速的 MELSECNET/10 或 MELSECNET/H 循环刷新。



- (4) COM 指令可在顺控程序中使用任意次。
- 但是，请注意，顺控程序的扫描周期将会因智能功能模块的一般数据处理和自动刷新（包括链接刷新）所花费的时间而增长。

备注

*1: 以下处理作为一般数据处理的一部分执行:

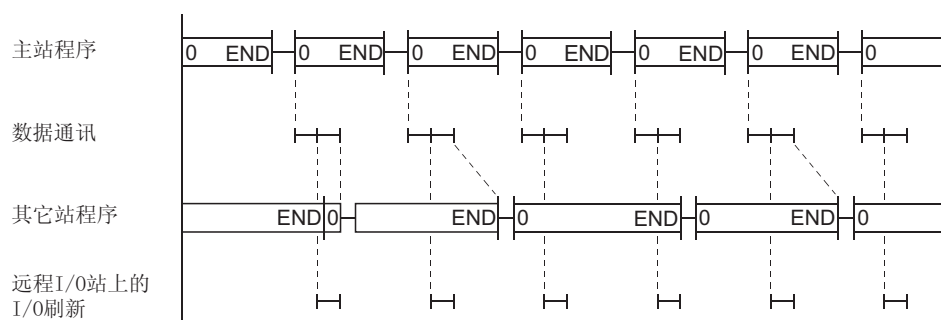
- 使用编程工具的通讯处理
- 其它站的监视处理
- 由别的智能功能模块的缓存的串行通讯模块进行的读取处理

*2: 自动刷新包括以下处理:

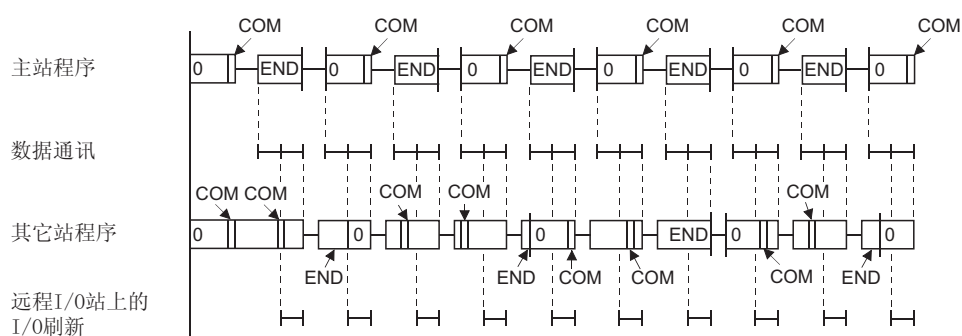
- 刷新 CC-Link
- 自动刷新智能功能模块
- 刷新 MELSECNET/10H

(5) 使用 COM 指令的数据通讯

(a) 未使用 COM 指令时的数据通讯示例



(b) 使用 COM 指令时的数据通讯示例

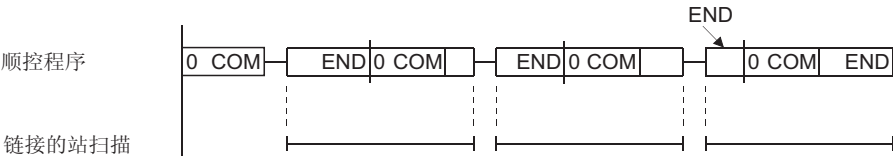


- ① 当在主站使用 COM 指令时，可以无条件提高同远程 I/O 站的数据通讯重复次数，如上面示例(b)所示，从而加快数据通讯的速度。
- ② 如果发生远程站顺控程序扫描周期比主站扫描周期长的情况，在主站使用 COM 指令将可以避免定时困难，在发生定时困难时数据将无法输入的，如上面示例(a)所示。

③ 当 COM 指令已经在其它站使用时，每次该站从主站接收到一个命令，就进行一次链接刷新。

- 第0步和COM指令之间
 - COM指令和COM指令之间
 - COM指令和END指令之间
- 每次在这些事件之间，
链接刷新可进行一次

(6) 如果来自链接站的扫描周期比主站的顺控程序扫描周期长，在主站指定 COM 指令将不会提高数据通讯速度。



要点	
COM 指令不可以用于低速执行类型程序、循环执行程序和中断程序。	

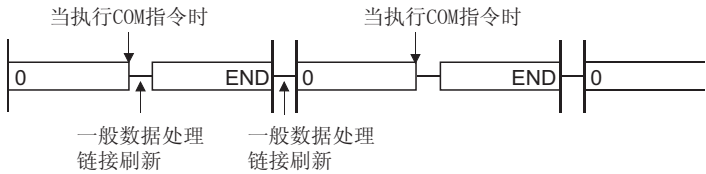
(2) 使用 QnACPU 时

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
—	—								



[功能]

- (1) 在以下情况下，使用 COM 指令：
- (a) 要提高同远程 I/O 站的传送/接收处理速度时。
 - (b) 要在执行数据链接期间，确保同使用不同扫描周期的其它站之间进行可靠的数据传送/接收。
- (2) 依据特殊继电器 SM775 是 ON 还是 OFF 状态，COM 指令的处理会有变化。
- 当 SM775 在 OFF 状态时：刷新链接，并进行一般的数据处理。*1
 - 当 SM775 在 ON 状态时：仅进行一般数据处理。*1
- (3) 在即将执行 COM 指令的时候，QnACPU 模块将暂停顺控程序的处理，而进行与 END 处理期间智能功能模块的一般数据处理和链接刷新相同的操作。



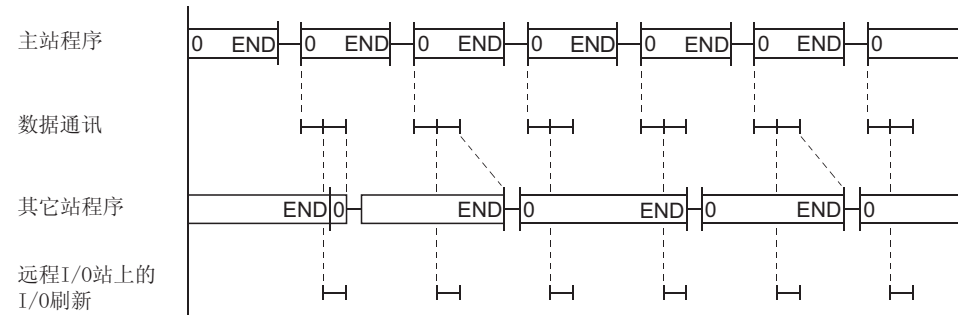
- (4) COM 指令可在顺控程序中使用任意次。
- 但是，请注意，顺控程序的扫描周期将会因智能功能模块的一般数据处理和链接刷新所花费的时间而增长。

备注

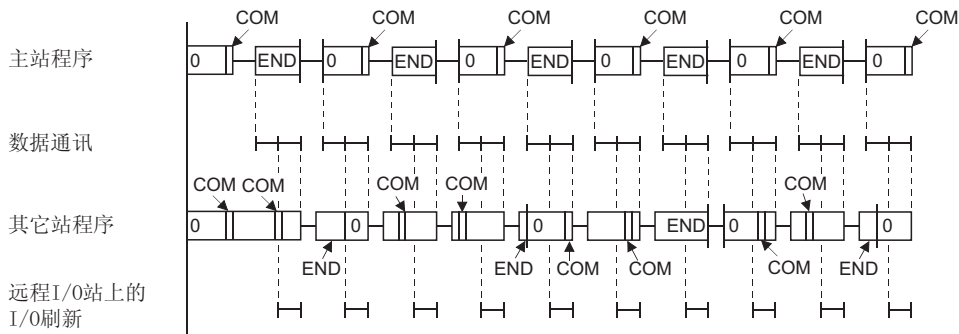
- 1) 以下处理作为一般数据处理的一部分执行:
- QnACPU 和外围设备之间的信号交换
 - 其它站的监视处理
 - 由别的智能功能模块的缓存的串行通讯模块进行的读取处理

(5) 使用 COM 指令的数据通讯

(a) 未使用 COM 指令时的数据通讯示例



(b) 已使用 COM 指令时的数据通讯示例



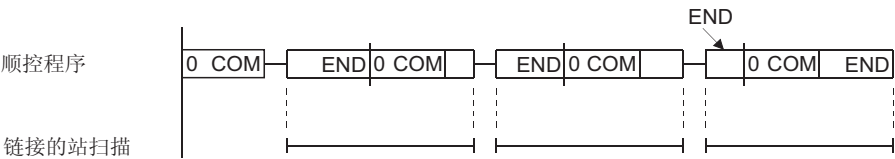
- ① 当在主站使用 COM 指令时，可以无条件提高同远程 I/O 站的数据通讯重复次数，如上面示例(b)所示，从而加快数据通讯的速度。
- ② 如果发生远程站顺控程序扫描周期比主站扫描周期长的情况，在主站使用 COM 指令将可以避免定时困难，在发生定时困难时数据将无法输入，如上面示例(a)所示。

③ 当 COM 指令已经在其它站使用时，每次该站从主站到接收一个命令，就进行一次链接刷新。

- 第0步和COM指令之间
 - COM指令和COM指令之间
 - COM指令和END指令之间

每次在这些事件之间，
链接刷新可进行一次

(6) 如果来自链接站的扫描周期比主站的顺控程序扫描周期长，在主站指定 COM 指令将不会提高数据通讯速度。



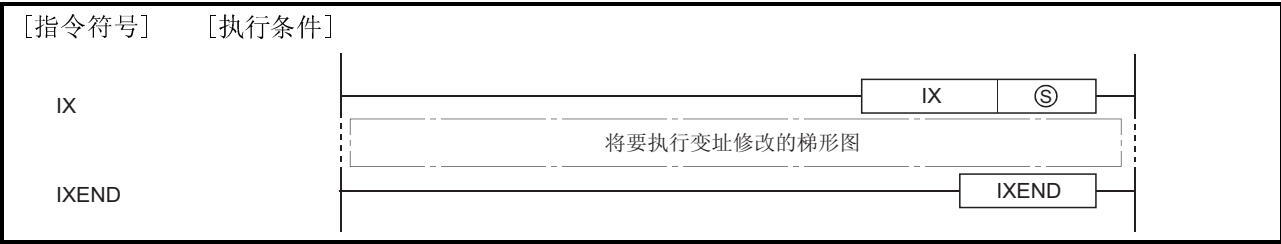
要点	
COM 指令不可以用于低速执行类型程序、循环执行程序和中断程序。	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.6.10 整个梯形图的变址修改(IX, IXEND)

(1) 使用基本型 QCPU 时

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][]G[][]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	—	○					—		

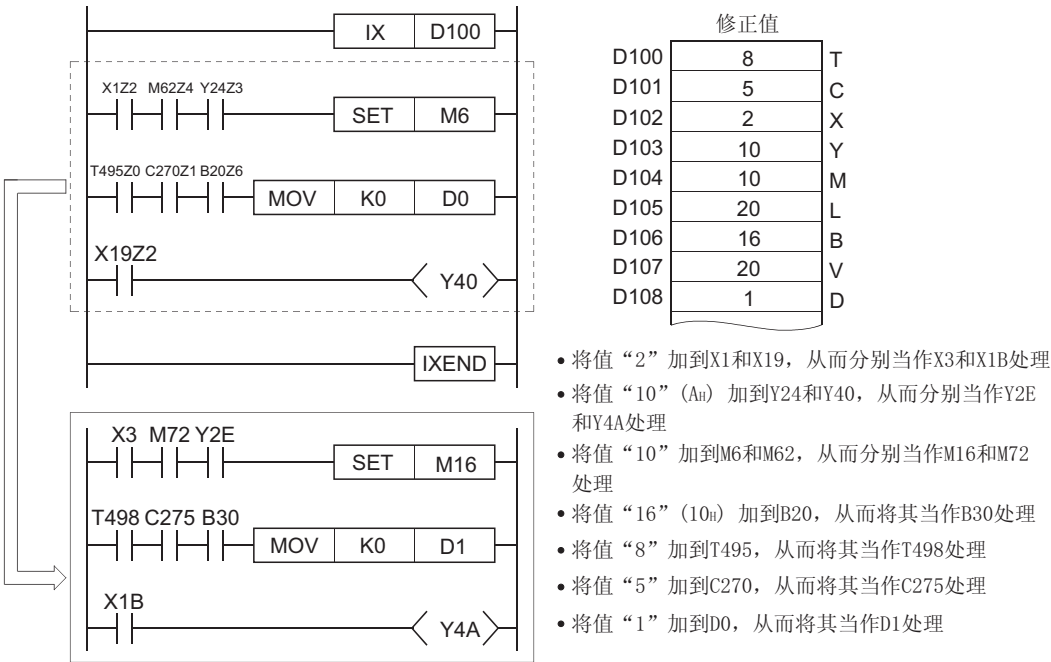


[设定数据]

设定数据	含义	数据类型
⑤	存储变址修改数据的软元件的起始软元件号	BIN 16 位

[功能]

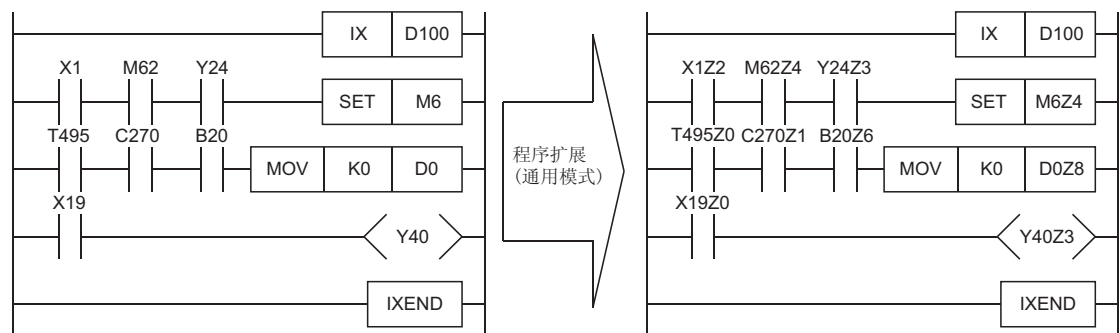
- (1) 为梯形图内从 IX 指令到 IXEND 指令的所有软元件进行软元件号变址修改。
- (2) 软元件号变址修改是通过为每个软元件预设修正值来实现的。将这些值加到梯形图内 IX 和 IXEND 指令之间所有使用的软元件上，并根据增加后的软元件号进行程序处理。



- (3) 那些在输入条件建立时仅执行一次的指令，诸如 PLS、PLF 和 $\boxed{}\boxed{}\boxed{}P$ 指令，无法通过使用 IX 到 IXEND 指令环进行变址修改。
- (4) 在增加修正值使得软元件号超出了软元件范围的情形下，无法进行准确处理。
- (5) 在顺控程序的在线程序改变期间，不要执行 IX 或 IXEND 指令（RUN 期间写入）。否则，无法进行准确处理。
- (6) 以 BIN 值为随机字软元件预设修正值，已经设定修正值的软元件的起始软元件号由 \textcircled{S} 指定。
- (7) 不要在 IX 和 IXEND 指令之间同时执行扫描执行类型程序和中断程序。

	软元件名称	索引寄存器号
\textcircled{S}	定时器修的正值(T)	Z0
$\textcircled{S} +1$	计数器的修正值(C)	Z1
$\textcircled{S} +2$	输入的修正值(X)	Z2
$\textcircled{S} +3$	输出的修正值(Y)	Z3
$\textcircled{S} +4$	内部继电器的修正值(M)	Z4
$\textcircled{S} +5$	锁存继电器的修正值(L)	Z5
$\textcircled{S} +6$	链接继电器的修正值(B)	Z6
$\textcircled{S} +7$	变址继电器的修正值(V)	Z7
$\textcircled{S} +8$	数据寄存器的修正值(D)	Z8
$\textcircled{S} +9$	链接寄存器的修正值(W)	Z9

(8) 应该为通过 IX 和 IXEND 指令建立的变址修改梯形图增加变址寄存器。
变址修改表的配置和对应的变址寄存器号如下所示：



＊ 以上程序中使用的“Zn”并不影响 IX/IXEND 指令以外的指令所使用的“Zn”。

要点
当在普通顺控程序和中断顺控程序中都使用 IX 和 IXEND 指令时，应设立互锁以避免同时执行。互锁假定普通顺控程序中的 IX 和 IXEND 指令之间的区域为 DI，即禁止中断。

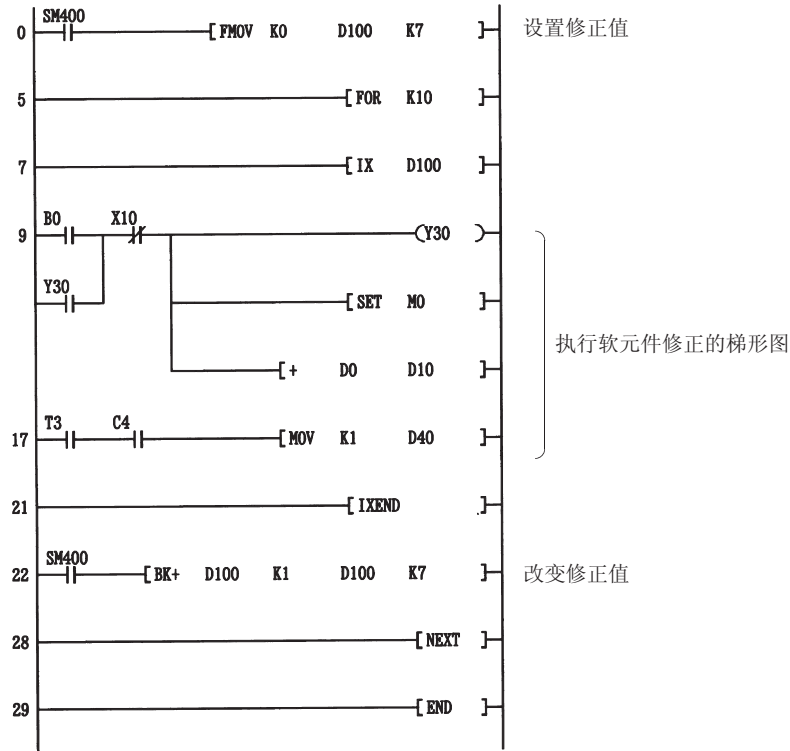
[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- IX 和 IXEND 指令未一起使用。 (错误代码：4231)
 - 在 IXEND 指令执行之前，且在执行 IX 指令之后，执行 END、FEND、GOEND 或者 STOP 指令。 (错误代码：4231)

[程序示例]

(1) 当改变软元件号时，以下程序执行相同梯形图 10 次。

[梯形图模式]



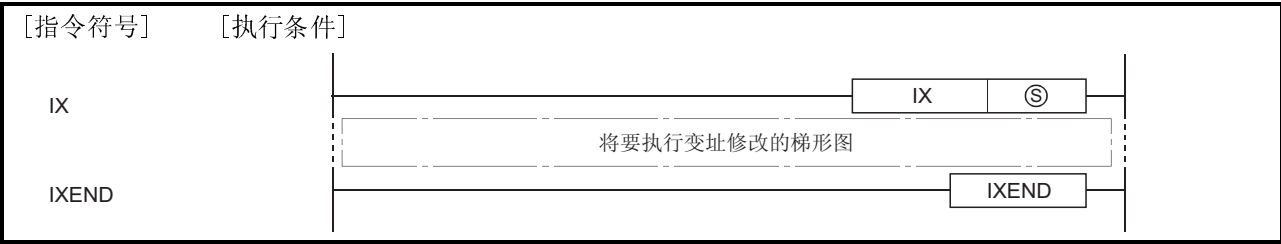
[列表模式]

步	指令	软元件
0	LD	SM400
1	FMOV	K0 D100 K7
5	FOR	K10
7	IX	D100
9	LD	B0
10	OR	Y30
11	ANI	X10
12	OUT	Y30
13	SET	M0
14	+	D0 D10
17	LD	T3
18	AND	C4
19	MOV	K1 D40
21	IXEND	
22	LD	SM400
23	BK+	D100 K1 D100 K7
28	NEXT	
29	END	

	修正值	第1次	第2次	第3次	第10次
D100	T修正值	B0 →	B1 →	B2 →	B9
D101	C修正值	X10 →	X11 →	X12 →	X19
D102	X修正值	Y30 →	Y31 →	Y32 →	Y39
D103	Y修正值	M0 →	M1 →	M2 →	M9
D104	M修正值	D0 →	D1 →	D2 →	D9
D105	L修正值	D10 →	D11 →	D12 →	D19
D106	B修正值	T3 →	T4 →	T5 →	T12
		C4 →	C5 →	C6 →	C13
		D40 →	D41 →	D42 →	D49

(2) 使用高性能型 QCPU/过程控制 CPU/QnACPU 时

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	—	○					—		

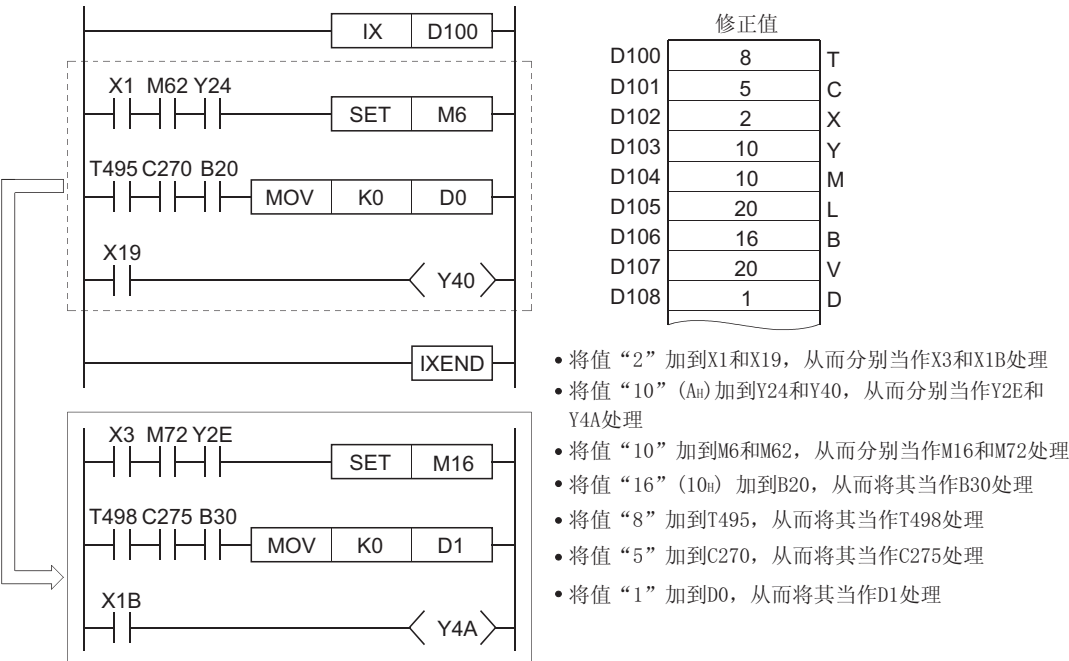


[设定数据]

设定数据	含义	数据类型
⑤	存储变址修改数据的软元件的起始软元件号	BIN 16 位

[功能]

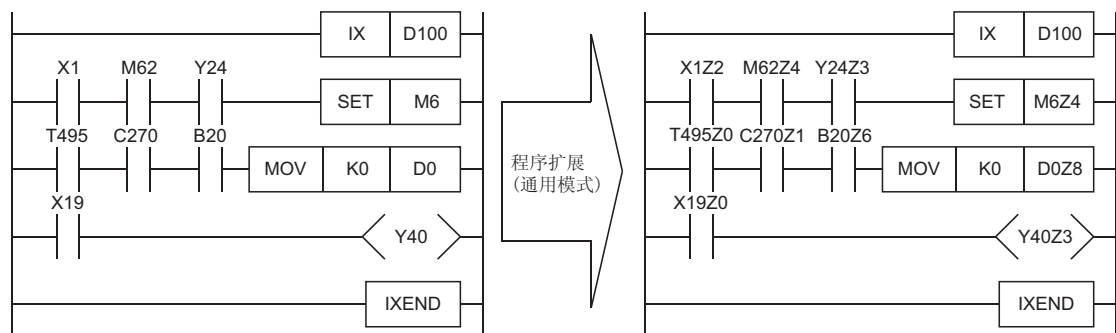
- (1) 为梯形图内从 IX 指令到 IXEND 指令的所有软元件进行软元件号变址修改。
- (2) 软元件号变址修改是通过为每个软元件预设修正值来实现的。这将这些值加到梯形图内 IX 和 IXEND 指令之间的所有使用的软元件上，根据增加后的软元件号进行程序处理。



- (3) 那些在建立输入条件时仅执行一次的指令，诸如 PLS、PLF 和 P 指令，无法通过使用 IX 到 IXEND 的指令环进行变址修改。
- (4) 在增加修正值使得软元件号超出了软元件范围的情形下，无法进行准确处理。
- (5) 在顺控程序的在线程序改变期间，不要执行 IX 或 IXEND 指令（RUN 期间写入）。
否则，无法进行准确处理。
- (6) 以 BIN 值为随机字软元件预设修正值，已经设定修正值的软元件的起始软元件号由⑤指定。
- (7) 不要在 IX 和 IXEND 指令之间同时执行扫描执行类型程序和中断程序。
- (8) 无法在 IX 和 IXEND 指令之间的程序中执行变址修改。
- (9) 不论是要扩展程序还是用户需要创建程序，都依赖于您的 GPP 功能软件包。
 - (a) 当扩展程序时(当使用 SW -GPPQ 时)
在程序扩展过程中，通过 IX 和 IXEND 指令建立的变址修改梯形图将被转换成使用变址寄存器 (Zn) 的梯形图。
变址修改表的配置和对应的变址寄存器号如下所示：

	软元件名称	变址寄存器号		软元件名称	变址寄存器号
⑤	定时器的修正值(T)	Z0	⑤ +8	数据寄存器的修正值(D)	Z8
⑤ +1	计数器的修正值(C)	Z1	⑤ +9	链接寄存器的修正值(W)	Z9
⑤ +2	输入的修正值(X)	Z2	⑤ +10	文件寄存器的修正值(R)	Z10
⑤ +3	输出的修正值(Y)	Z3	⑤ +11	缓冲寄存器I/O号的修正值(U)	Z11
⑤ +4	内部继电器的修正值(M)	Z4	⑤ +12	缓冲寄存器的修正值(G)	Z12
⑤ +5	锁存继电器的修正值(L)	Z5	⑤ +13	链接直接软元件网络号的修正值(J)	Z13
⑤ +6	链接继电器的修正值(B)	Z6	⑤ +14	文件寄存器的修正值(ZR)	Z14
⑤ +7	变址继电器的修正值(V)	Z7	⑤ +15	指针的修正值(P)	Z15

(b) 用户需要创建程序时(使用 GX Developer 时)
应该给通过 IX 和 IXEND 指令建立的变址修改梯形图增加变址寄存器



* 以上程序中使用的“Zn”并不影响 IX/IXEND 指令以外的指令所使用的“Zn”。

要点
(1) 当使用 GPPQ(仅适用于 QnCPU)编辑程序时，必须在“通用模式”下启动 GPPQ，并进行“程序扩展”，以便使用 IX 和 IXEND 指令创建变址修改梯形图。 当用 CPU 模块型号(Q2A, Q2A-S1, Q2AS(H), Q2AS-S1, Q2AS(H)-S1, Q3A, Q4A or Q4AR)启动 GPPQ 以通过 IX 和 IXEND 指令创建变址修改梯形图时，则无法进行正常处理。
(2) 当在普通顺控程序和中断顺控程序中都使用 IX 和 IXEND 指令时，应设立互锁以避免同时执行。互锁假定普通顺控程序中的 IX 和 IXEND 指令之间的区域为 DI，即禁止中断。

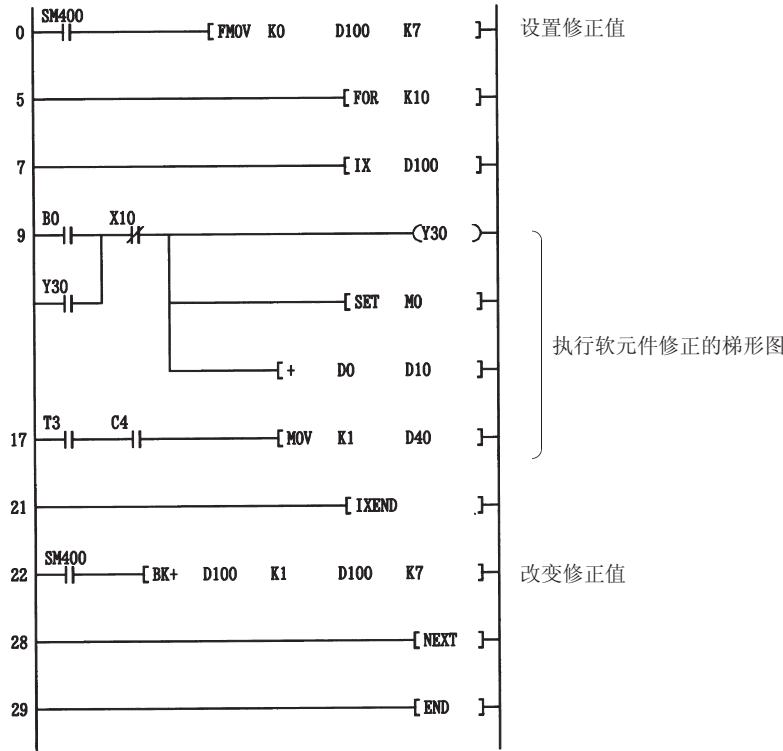
[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- IX 和 IXEND 指令未一起使用。(错误代码: 4231)
 - 在 IXEND 指令执行之前，且在执行 IX 指令之后，执行 END、FEND、GOEND 或者 STOP 指令。(错误代码: 4231)

[程序示例]

(1) 当改变软元件号时，以下程序执行相同梯形图 10 次。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	FMOV	K0 D100 K7
5	FOR	K10
7	IX	D100
9	LD	B0
10	OR	Y30
11	ANI	X10
12	OUT	Y30
13	SET	M0
14	+	D0 D10
17	LD	T3
18	AND	C4
19	MOV	K1 D40
21	IXEND	
22	LD	SM400
23	BK+	D100 K1 D100 K7
28	NEXT	
29	END	

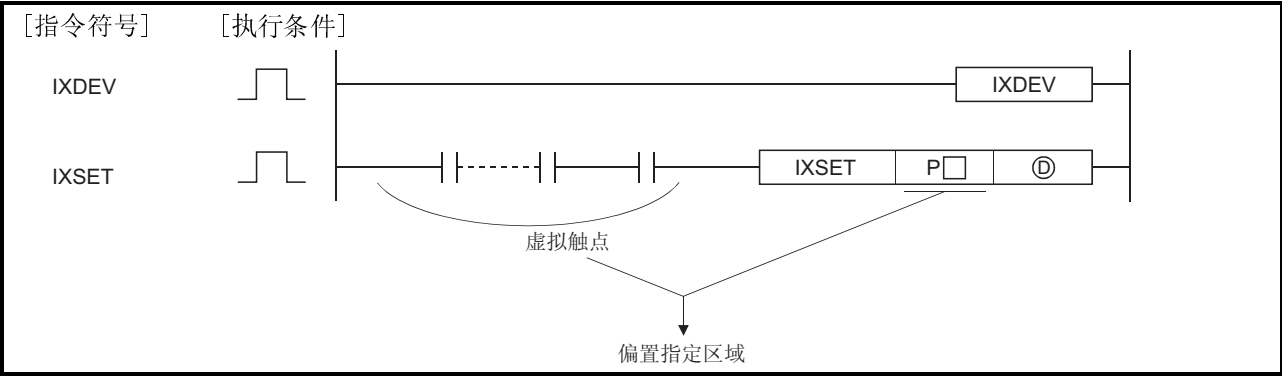
	修正值	第1次	第2次	第3次	第10次
D100	T修正值	B0 →	B1 →	B2 →	B9
D101	X修正值	X10 →	X11 →	X12 →	X19
D102	Y修正值	Y30 →	Y31 →	Y32 →	Y39
D103	M修正值	M0 →	M1 →	M2 →	M9
D104	D修正值	D0 →	D1 →	D2 →	D9
D105	L修正值	T3 →	T4 →	T5 →	T12
D106	C修正值	C4 →	C5 →	C6 →	C13
	B修正值	D40 →	D41 →	D42 →	D49

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.6.11 整个梯形图的变址修改中修正值的指定 (IXDEV, IXSET)

(1) 使用基本型 QCPU 时

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\C[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				P
P[]	—	—		—				○	
⑤	—	○		—				—	



[设定数据]

设定数据	含义	数据类型
P[]	存储变址修改数据的软元件（仅指针）的起始软元件号	软元件名称
Ⓓ	存储变址修改数据的软元件（除指针之外）的起始软元件号	BIN 16 位

[功能]

- (1) 在Ⓓ指定的变址修改表上设定在偏置指定区域指定的软元件偏置值。
- (2) 如果未进行值指定，则输入 0 值。
- (3) 字软元件也通过触点（字软元件位指定）显示。
通过 D10.0 指定数据寄存器 10 (D10)。
(从 0 到 F 的任何值都可用作位号。)

- (4) 根据以下阐述的方法进行指定。
(符号□代表偏置值，记号 XX 代表随机选择。)

软元件	T	C	X	Y	M	L	V	B
指定方法	$T\begin{smallmatrix}\square\\ \\ \\\hline\end{smallmatrix}$	$C\begin{smallmatrix}\square\\ \\ \\\hline\end{smallmatrix}$	$X\begin{smallmatrix}\square\\ \\ \\\hline\end{smallmatrix}$	$Y\begin{smallmatrix}\square\\ \\ \\\hline\end{smallmatrix}$	$M\begin{smallmatrix}\square\\ \\ \\\hline\end{smallmatrix}$	$L\begin{smallmatrix}\square\\ \\ \\\hline\end{smallmatrix}$	$V\begin{smallmatrix}\square\\ \\ \\\hline\uparrow\\\hline\end{smallmatrix}$	$B\begin{smallmatrix}\square\\ \\ \\\hline\end{smallmatrix}$
软元件	D	W	P					
指定方法	$D\begin{smallmatrix}\square\\ \\ \\\hline\end{smallmatrix}.XX$	$W\begin{smallmatrix}\square\\ \\ \\\hline\end{smallmatrix}.XX$	虚拟					

- (5) 如果在偏置指定区域已经为两个相同类型的软元件设定了两个偏置，则最后设置的值有效。
- (6) 应将 IXDEV 和 IXSET 指令视为一对指令。
- (7) 任何从 0 到 32767 之间的 ZR 值都有效。
(偏置值是指定的软元件号除以 23768 所得的余数。)
- (8) 仅 LD 和 AND 对偏置指定区域的虚拟触点有效，此处使用的任何其它指令都将被忽略。

[运行错误]

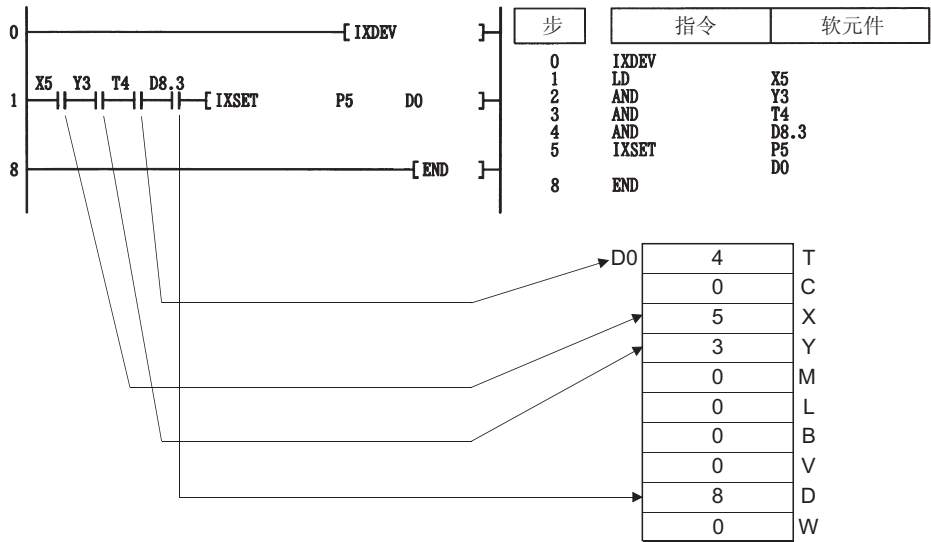
- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- IXDEV 和 IXSET 指令未成对使用。(错误代码：4231)

[程序示例]

- (1) 以下程序为输入(X)、输出(Y)、数据寄存器(D)和指针(P)更改修正值。

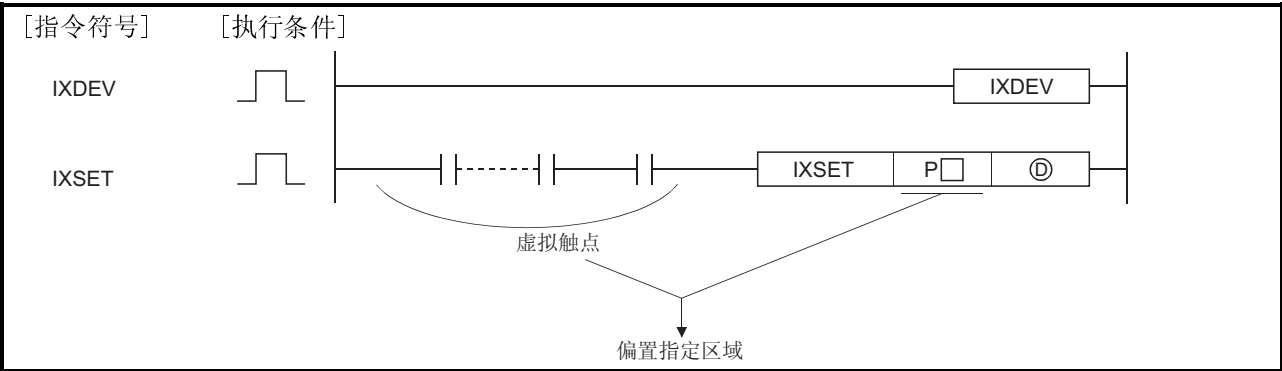
[梯形图模式]

[列表模式]



(2) 使用高性能型 QCPU/过程控制 CPU/QnACPU 时

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\C[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				P
P[]	—	—		—					○
⑤	—	○		—					—



[设定数据]

设定数据	含义	数据类型
P[]	存储变址修改数据的软元件（仅指针）的起始软元件号	软元件名称
Ⓓ	存储变址修改数据的软元件（除指针之外）的起始软元件号	BIN 16 位

[功能]

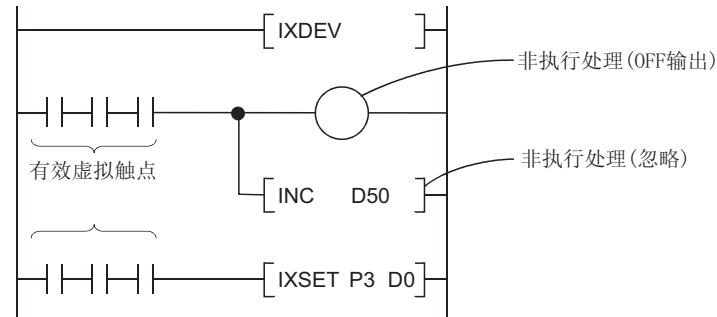
- (1) 在Ⓓ指定的变址修改表上设定在偏置指定区域指定的软元件偏置值。
- (2) 如果未进行值指定，则输入 0 值。
- (3) 字软元件也通过触点（字软元件位指定）显示。
通过 D10.0 指定数据寄存器 10 (D10)。
(从 0 到 F 的任何值都可用作位号。)

- (4) 根据以下阐述的方法进行指定。
- (符号□代表偏置值，记号 XX 代表随机选择。)

软元件	T	C	X	Y	M	L	V	B
指定方法	<div>T□ ┌┴┐</div>	<div>C□ ┌┴┐</div>	<div>X□ ┌┴┐</div>	<div>Y□ ┌┴┐</div>	<div>M□ ┌┴┐</div>	<div>L□ ┌┴┐</div>	<div>V□ ┌┴┐ ↑</div>	<div>B□ ┌┴┐</div>
软元件	D	W	R	U/G		J		ZR
指定方法	<div>D□.XX ┌┴┐</div>	<div>W□.XX ┌┴┐</div>	<div>R□.XX ┌┴┐</div>	<div>U□\G□.XX ┌┴┐</div>		<div>J□\B□[*] ┌┴┐</div>		<div>ZR□.XX ┌┴┐</div>
软元件	P	*:J □ □ \ 之后的软元件指定 B, W, X, 或 Y, 且偏置值也被设置为与此相对应。						
指定方法	通过软元件指定设置 P□							

- (5) 如果在偏置指定区域已经为两个相同类型的软元件设定了两个偏置，则最后设置的值有效。
- (6) 应将 IXDEV 和 IXSET 指令视为一对指令。
- (7) 任何从 0 到 32767 之间的 ZR 值都有效。
- (偏置值是指定的软元件号除以 23768 所得的余数。)
- (8) 在偏置指定部位的虚拟触点仅对位于 IXDEV-IXSET 指令范围内的 LD 和 AND 有效，如果指定了其它指令，将不执行 IXDEV-IXSET 指令。

示例)



[运行错误]

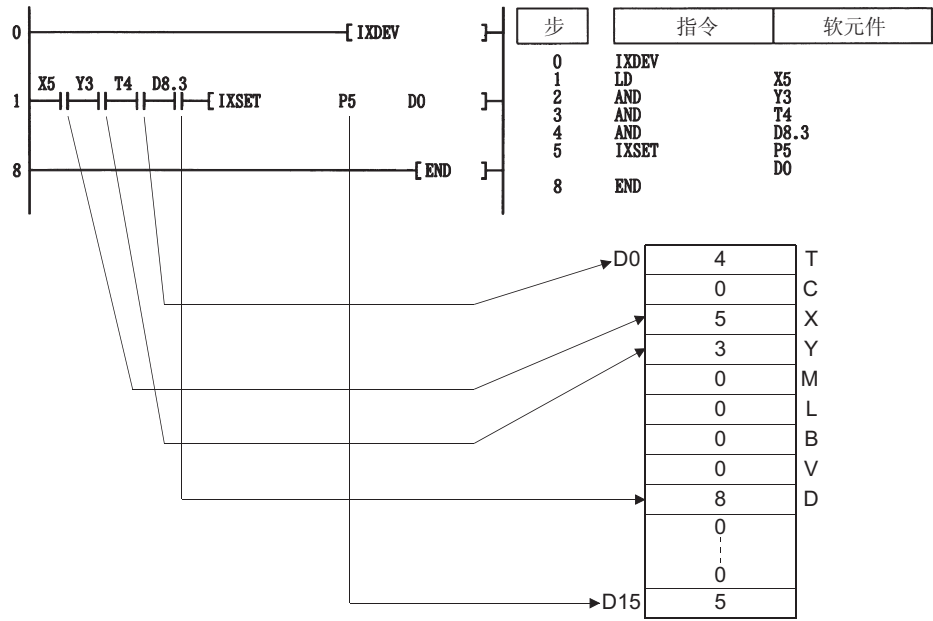
- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- IXDEV 和 IXSET 指令未成对使用。 (错误代码: 4231)

[程序示例]

(1) 以下程序为输入(X)、输出(Y)、数据寄存器(D)和指针(P)改变修正值。

[梯形图模式]

[列表模式]

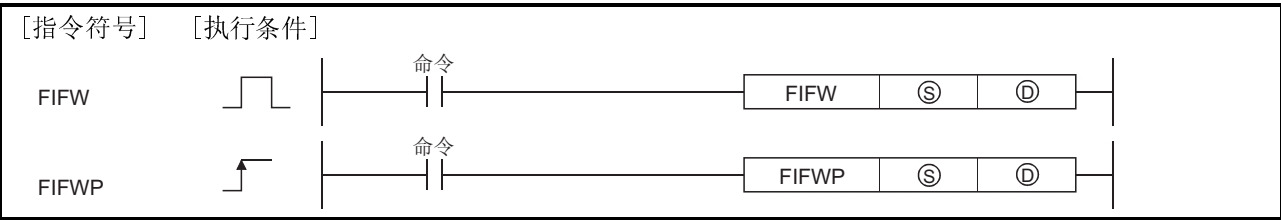


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.7 数据表格操作指令

7.7.1 将数据写入数据表格 (FIFW, FIFWP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○	○			○			○	—
⑥	—	○			—			—	—

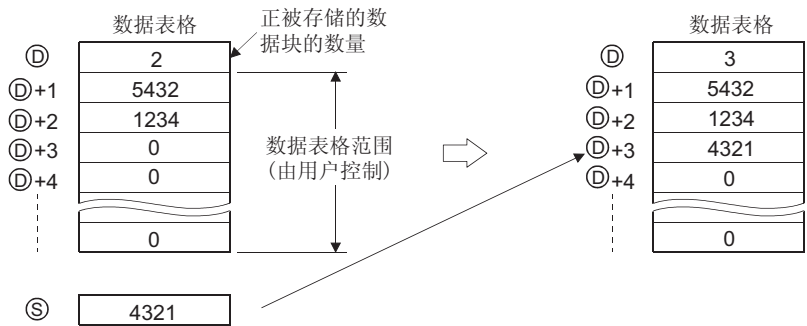


[设定数据]

设定数据	含义	数据类型
⑤	要写入表格中的数据或存储该数据的软元件号	BIN 16 位
⑥	表格的起始号	

[功能]

- (1) 将⑤指定的 16 位数据存储在⑥指定的数据表格中。
在表格中存储的数据块的数量存储在⑥上, 且将⑤指定的数据按照顺序从⑥+1 处开始存储。



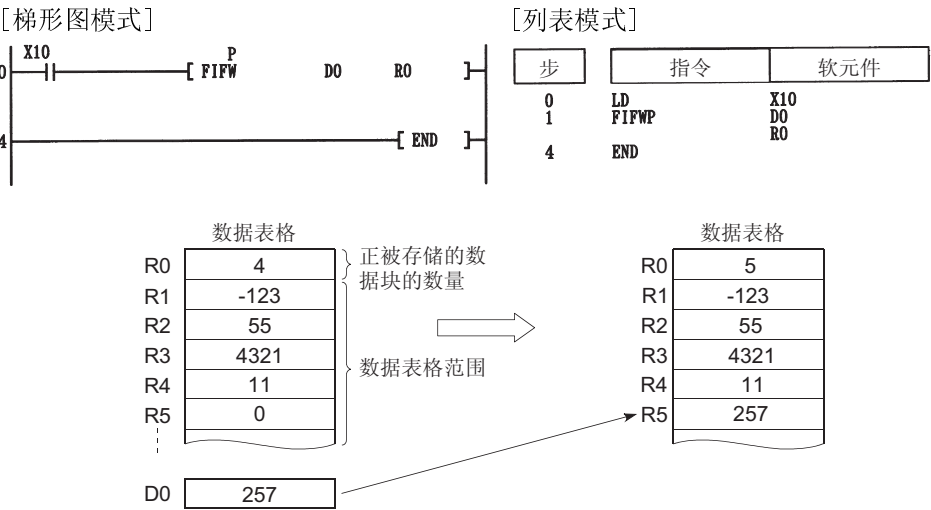
- (2) 第一次执行 FIFW 指令时, 应将指定软元件中的所有值都清除。
- (3) 将要写入数据表格的数据块的数量和数据表格范围应由用户控制。(参见程序示例(2))

[运行错误]

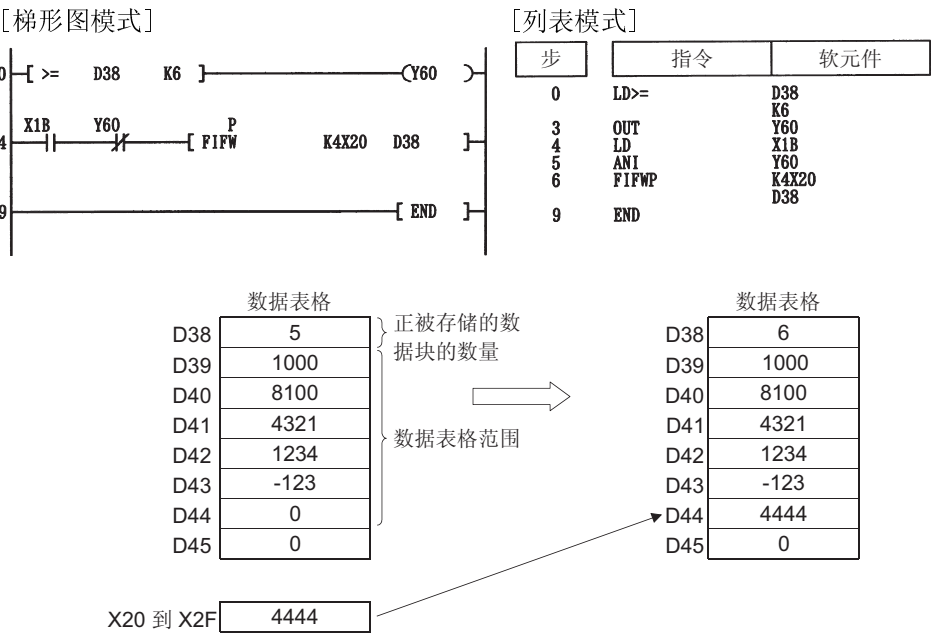
- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，且错误代码存储在 SD0 上。
- 执行 FIFW 指令时，数据表格范围超出了相关软元件范围。 (错误代码:4101)

[程序示例]

- (1) 当 X10 为 ON 时，以下程序将数据存储在 R0 之后的 D0 上。



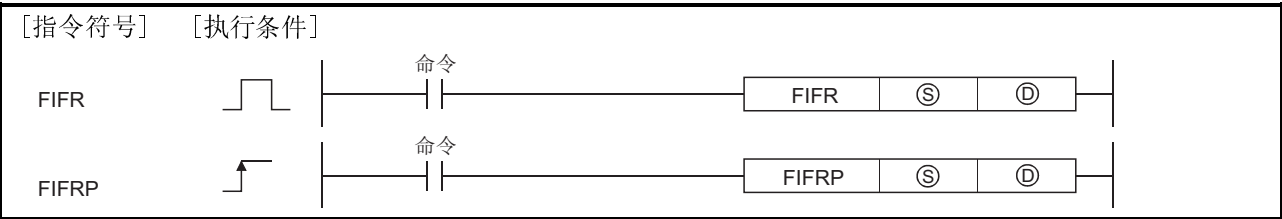
- (2) 当 X1B 为 ON 时，以下程序将 X20 到 X2F 的数据存储在 D38 到 D44 上，并且，如果有 6 个以上的数据块要存储，则将 Y60 变为 ON，禁用 FIFW 指令。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.7.2 从表格中读取最旧的数据 (FIFR, FIFRP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [J][K]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	○	○			○		—
⑥	—	○			—		—

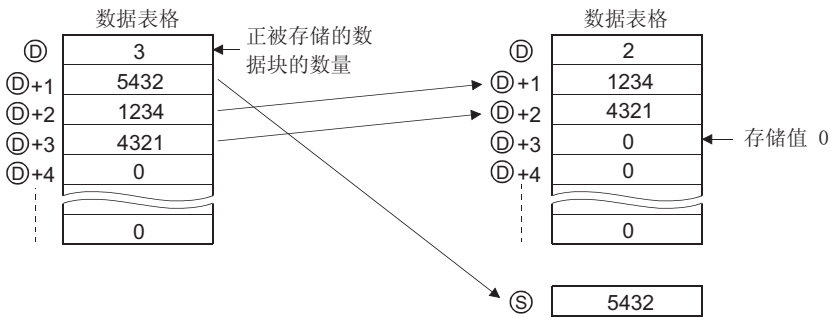


[设定数据]

设定数据	含义	数据类型
⑤	用于存储从表格中读取的数据的软元件的起始号	BIN 16 位
⑥	起始表格号	

[功能]

- (1) 将输入到由⑥指定的表格的最旧数据 (⑥+1) 存储在⑤指定的软元件上。
在执行 FIFR 指令之后，表格中的数据被向上压缩一个数据块。



- (2) 如果存储在⑥上的值为 0，用户应努力避免执行 FIFR 指令。(参见程序示例(1))

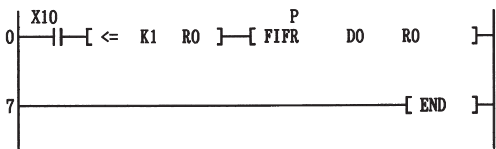
[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，且错误代码存储在 SD0 上。
- 当⑤上的值为 0 时执行了 FIFR 指令。(错误代码:4100)
 - 执行 FIFW 指令时，数据表格范围超出了相应软元件范围。(错误代码:4101)

[程序示例]

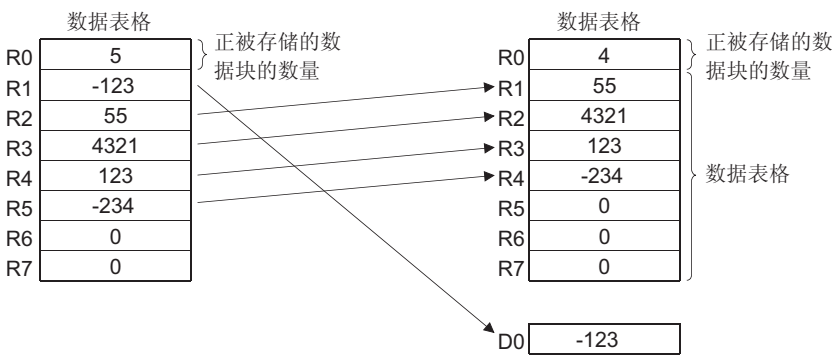
(1) 当 X10 变为 ON 时，以下程序将占用 R0 到 R7 的数据表格中的在 R1 上保持的数据存储到目的点 D0 上。

[梯形图模式]



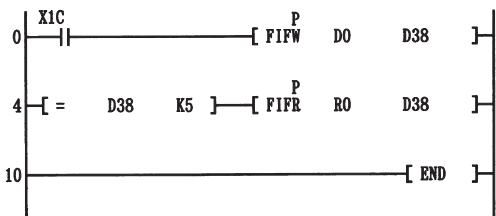
[列表模式]

步	指令	软元件
0	LD	X10
1	AND<=	K1
		R0
4	FIFRP	DO
		R0
7	END	



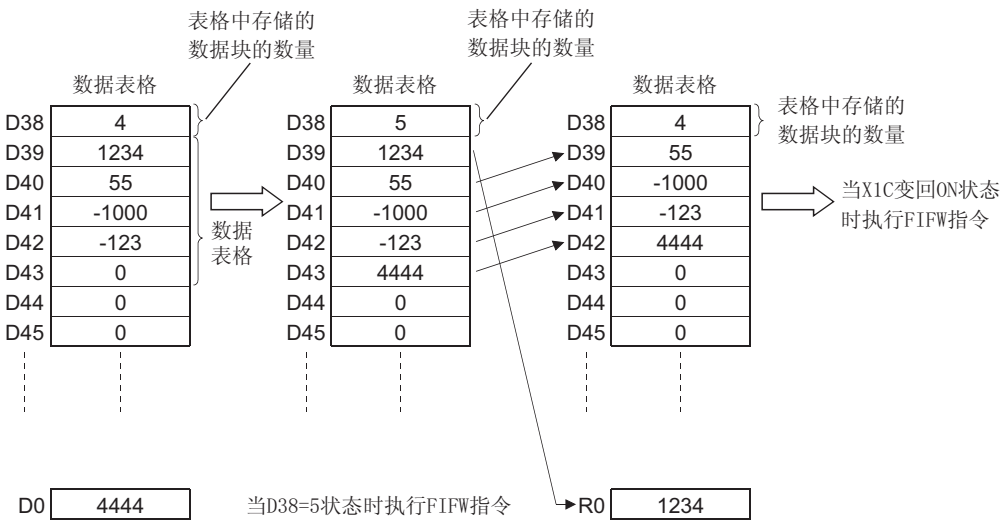
(2) 当 X1C 变为 ON 时，以下程序将 D0 中的数据存储在占用 D38 到 D43 的数据表格中，并且当表格有 5 个数据项时，将数据表格中在 D39 上保持的数据存储到 R0 上。

[梯形图模式]



[列表模式]

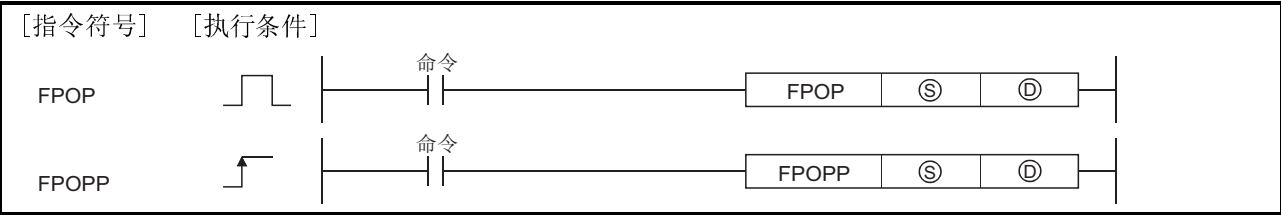
步	指令	软元件
0	LD	X1C
1	FIFWP	D0
		D38
4	LD=	D38
		D38
7	FIFRP	K5
		R0
		D38
10	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.7.3 从数据表格中读取最新数据(FPOP, FPOPP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量	其它
	位	字		位	字				
⑤	○	○			○				—
⑥	—	○			—				—

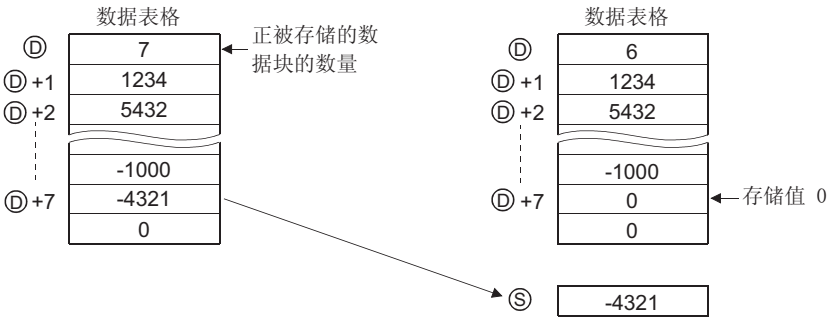


[设定数据]

设定数据	含义	数据类型
⑤	用于存储从表格中读取的数据的软元件的起始号	BIN 16 位
⑥	起始表格号	

[功能]

- (1) 将输入到⑥指定的表格的最新数据存储到⑤指定的软元件上。
在执行 FPOP 指令之后，保存由 FPOP 指令读取的数据的软元件被复位到 0。



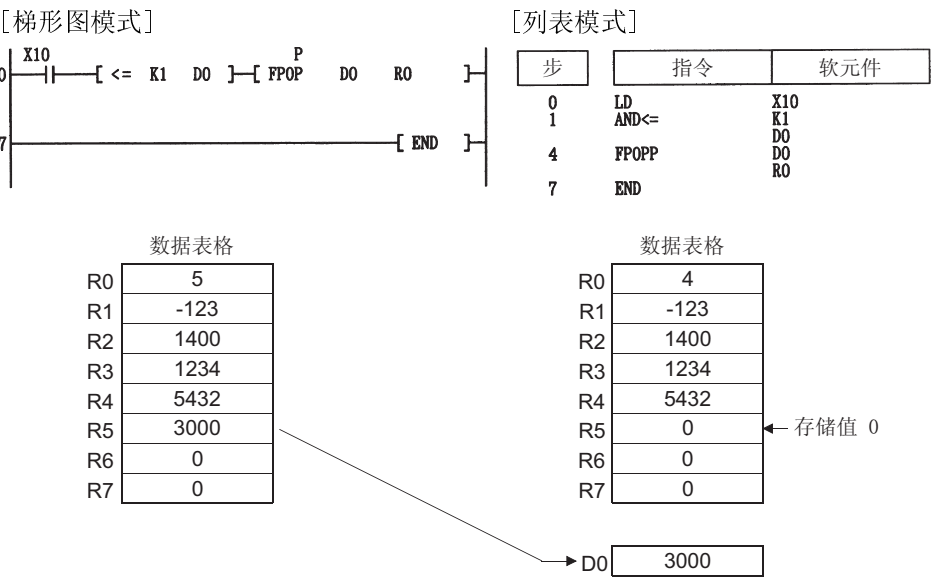
- (2) 如果存储在⑥上的值为 0，用户应努力避免执行 FPOP 指令。(参见程序示例(1))

[运行错误]

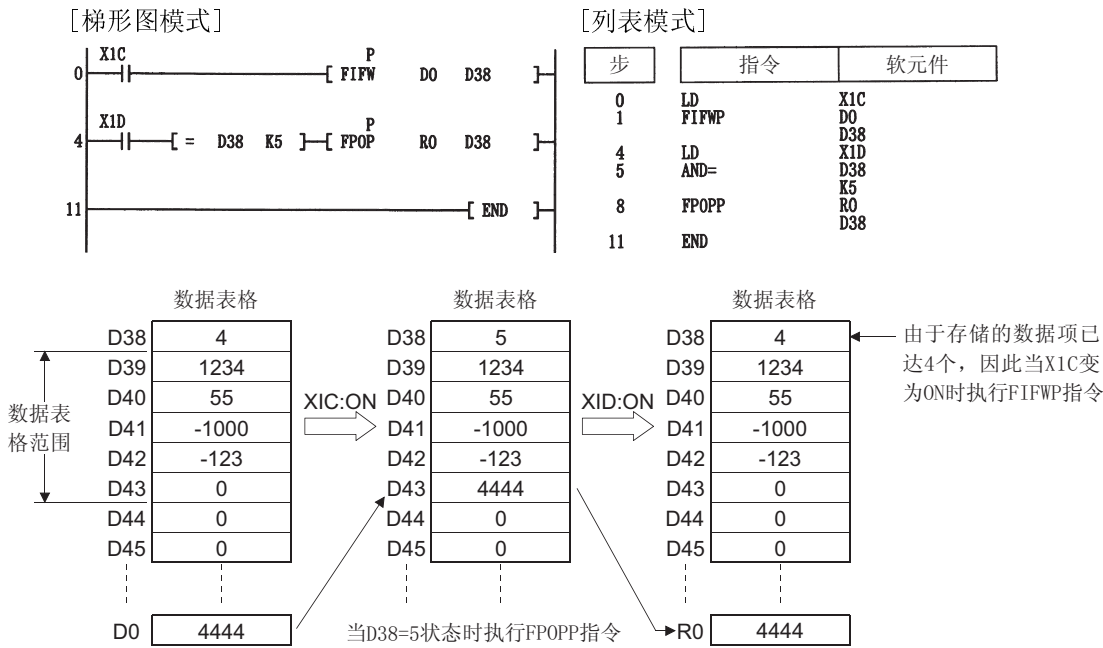
- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，且错误代码存储在 SD0 上。
- 当⑥值为 0 时，执行了 FPOP 指令。(错误代码: 4100)
 - 执行 FPOP 指令时，数据表格范围超出了对应软元件的范围。(错误代码: 4101)

[程序示例]

(1) 当 X10 为 ON 时，以下程序将占用 R0 到 R7 的数据表格中存储的更新数据存储到 D0 上。



(2) 当 X1C 变为 ON 时，以下程序将原先存储在 D0 中的数据存储到占用 D38 到 D43 的数据表格中，并且当表格中的数据项数达到 5 时，将 X1D 变为 ON，将存放在数据表格中的更新数据存储到 R0 上。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7. 7. 4 从数据表格中删除数据和在数据表格中插入数据
(FDEL, FDELP, FINS, FINSP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
Ⓔ	○	○		○				—	—
Ⓕ	—	○		—				—	—
n	○	○		○				○	—

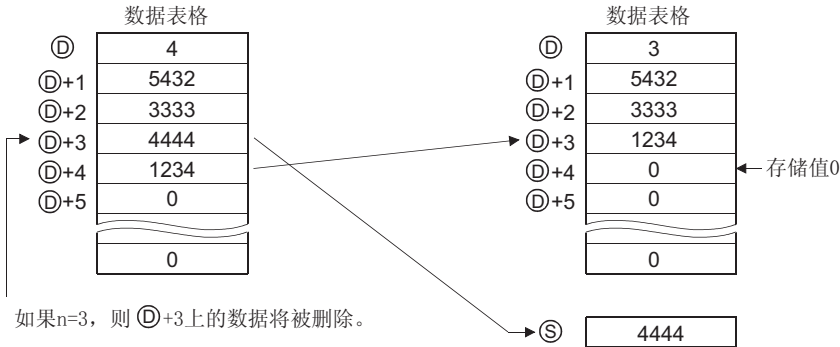
[指令符号]	[执行条件]	□ 表示 FDEL/FINS			
FDEL		命令	□	Ⓢ	Ⓓ n
FINS					
FDELP		命令	□ P	Ⓢ	Ⓓ n
FINSP					

[设定数据]

设定数据	含义	数据类型
Ⓢ	插入的数据或用于存储插入数据的软元件的起始号 用于存储删除数据的软元件的起始号	BIN 16 位
Ⓓ	表格的起始号	
n	表格中存放插入或删除数据的位置	

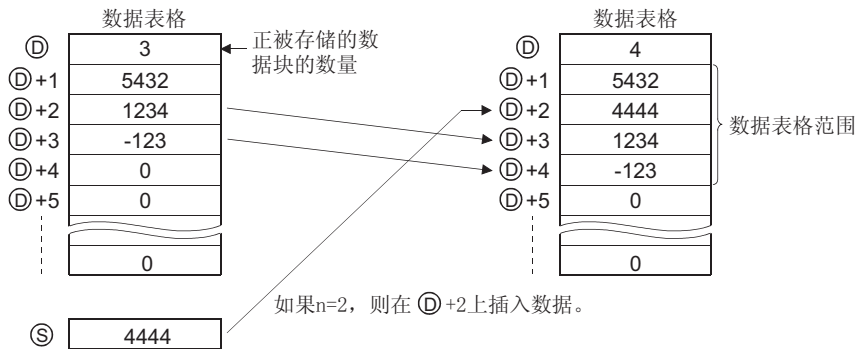
[功能]

- FDEL
- (1) 从Ⓓ指定的数据表格中删除第 n 个数据块，并将其存储到Ⓢ指定的软元件上。从Ⓓ指定的数据表格中删除第 n 个数据块，并将其存储到Ⓢ指定的软元件上。
在执行 FDEL 指令之后，表格中被删除的数据块之后的数据被向前压缩一个数据块。



FDEL

- (1) 将⑤指定的 16 位数据插入到由④指定的数据表格的第 n 个数据块上。
在执行 FINS 指令之后，表格中在插入的数据块之后的数据被全部下调一个位置。

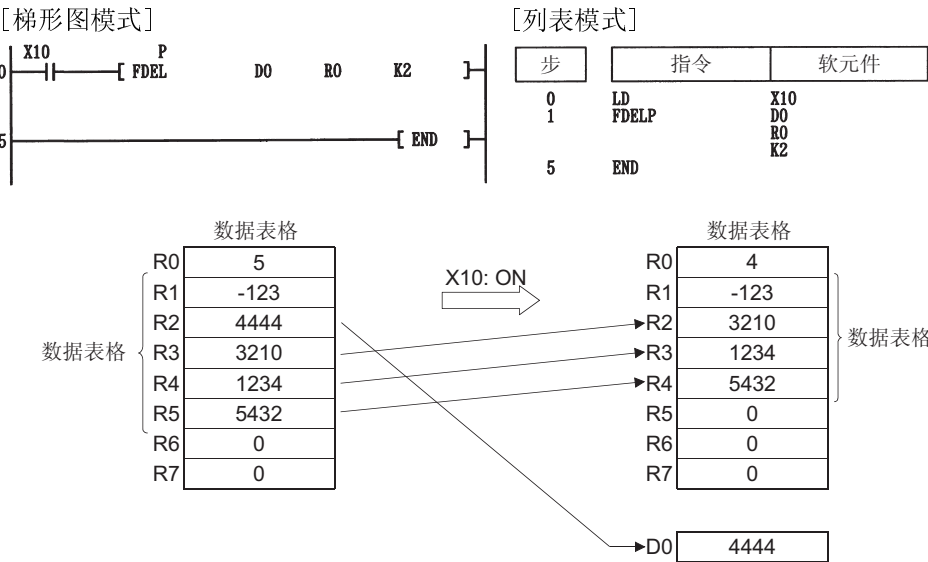


[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，且错误代码存储在 SD0 上。
- 在使用 FDEL、FINS 指令的情况下，从④开始的第 n 个位置，大于表格中实际数据块的数量加 1。 (错误代码:4101)
 - 在使用 FDEL、FINS 指令的情况下，值 n 超出了表格④的软元件范围。 (错误代码:4101)
 - 当 n=0 时，执行了 FDEL 或 FINS 指令。 (错误代码:4100)
 - 当④的值为 0 时，执行了 FDEL 指令。 (错误代码:4100)
 - 在执行 FDEL 或 FINS 指令时，数据表格范围超出了对应软元件的范围。 (错误代码:4101)

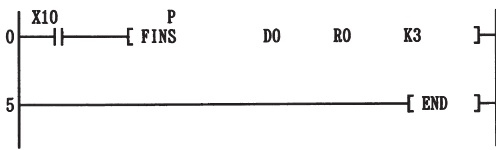
[程序示例]

- (1) 当 X10 变为 ON 时，以下程序将从占用 R0 到 R7 的表格的第 2 个位置删除数据，并将此数据存储到 D0 上。



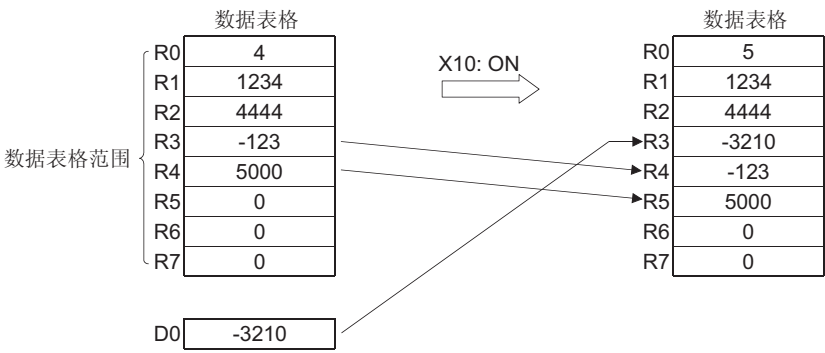
(2) 当 X10 为 ON 时，以下程序将 D0 上的数据插入到占用 R0 到 R7 的表格的第 3 个数据块上。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X10
1	FINSP	D0 R0 K3
5	END	

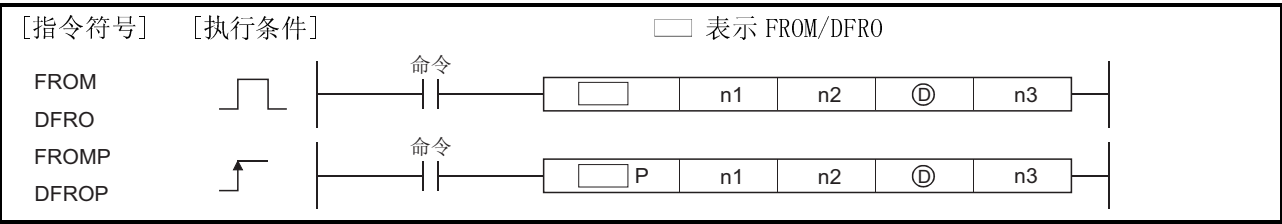


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.8 缓冲存储区访问指令

7.8.1 从智能功能模块/特殊功能模块中读取 1 字/2 字数据
(FROM, FROMP, DFRO, DFROP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J□□□		特殊功能模 块 U□□G□□	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				U
n1	○			○				○	○
n2	○			○				○	—
Ⓓ	○			—				○	—
n3	○			○				○	—



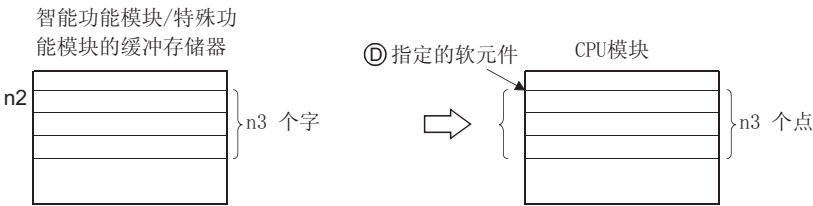
[设定数据]

设定数据	含义	数据类型
n1	智能功能模块/特殊功能模块的起始 I/O 号	BIN 16 位
n2	被读取数据的起始地址	
Ⓓ	存储被读取数据的软元件的起始号	BIN 16/32 位
n3	被读取数据的数量 ● FROM(P) : 1 到 6144 ● DFRO(P) : 1 到 3072 (仅适用于 A/QnA)	BIN 16 位

[功能]

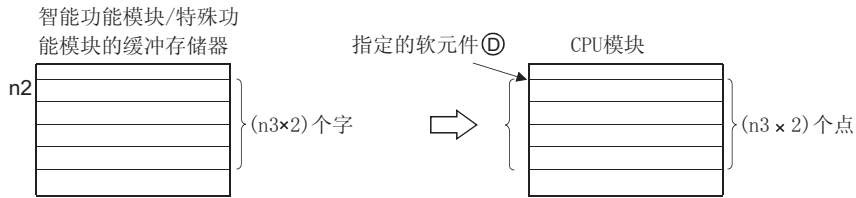
FROM

- (1) n1 指定的智能功能模块/特殊功能模块的缓冲存储器中，从由 n2 指定的地址开始读取 n3 个字的数据。
然后，将数据存储在从Ⓓ指定的软元件开始的区域。



DFR0

(1) 在 n1 指定的智能功能模块/特殊功能模块的缓冲存储器中，从 n2 指定的地址开始读取 (n3×2) 字的数据。并将数据存储在与④指定的软元件开始的区域。



要点

- (1) 智能功能模块/特殊功能模块的软元件也可以用于从智能功能模块/特殊功能模块软元件的缓冲存储器上读取数据。
智能功能模块/特殊功能模块软元件的详细情况，请参考使用的 CPU 模块的用户手册(功能解释, 编程基础)或 QnACPU 编程手册(基础)。

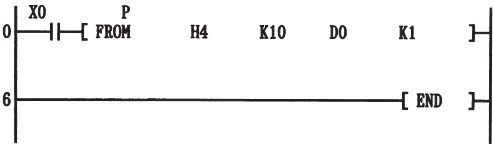
[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志(SM0)变为 ON，且错误代码存储在 SD0 上。
- 在执行指令时，没有同智能功能模块/特殊功能模块进行信号交换。(错误代码:1412)
 - 在执行指令时，智能功能模块/特殊功能模块上检测到错误。(错误代码:1402)
 - n1 指定的 I/O 号不是用于智能功能模块/特殊功能模块。(错误代码:2110)
 - ④指定的软元件的 n3 (对于 DFR0 是 2×n3 个点) 个点超过了指定软元件的范围。(错误代码:4101)
 - n2 指定的地址超出了缓冲存储器的范围。(错误代码:4100)
 - n2 指定的地址是一个奇数地址。(AJ71QC24(N)) (错误代码:4100)
 - 读取数据的数量超出了 6144/3072。(仅适用于 A/QnA 模块) (错误代码:4101)

[程序示例]

(1) 当 X0 变为 ON 时，以下程序将安装在 I/O 号 040 到 05F 的 A68AD 的 CH1 数字值读到 D0 上。(从缓冲存储器地址 10 中读取 1 个字的数据)。

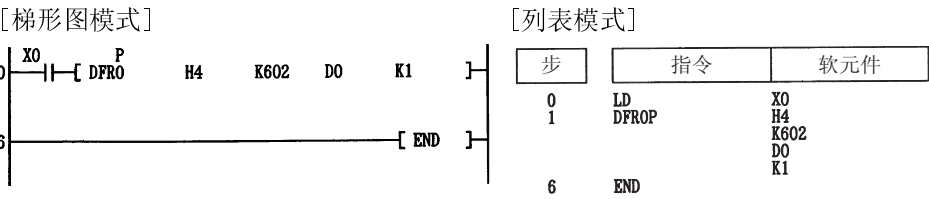
[梯形图模式]



[列表模式]

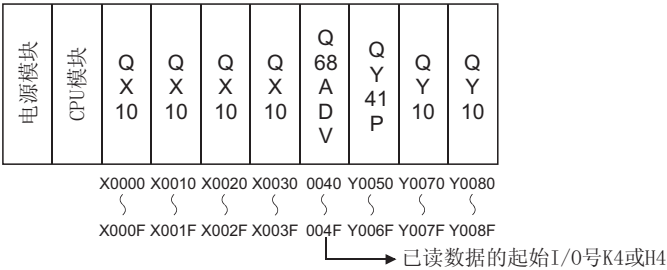
步	指令	软元件
0	LD	X0
1	FROMP	H4 K10 D0 K1
6	END	

(2) 当 X0 变为 ON 时，以下程序将安装在 I/O 号 040 到 05F 的 AD71 的 X 轴的当前值读取到 D0 和 D1 上。（从缓冲存储器地址 602, 603 读取 2 字的数据）。



备注

1) 当以十六进制 4 位数表示安装智能功能模块/特殊功能模块的插槽的起始 I/O 号时，n1 的值由高三位指定。

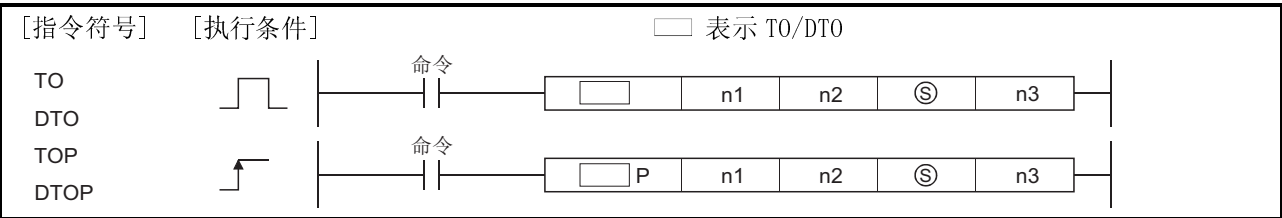


2) QCPU 建立了 FROM/DFRO 指令的自动互锁。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.8.2 将 1 字/2 字数据写入智能功能模块/特殊功能模块(T0, TOP, DT0, DTOP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[] []		特殊功能模 块 U[] \ G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				U
n1	○			○				○	○
n2	○			○				○	—
Ⓢ	○			—				○	—
n3	○			○				○	—

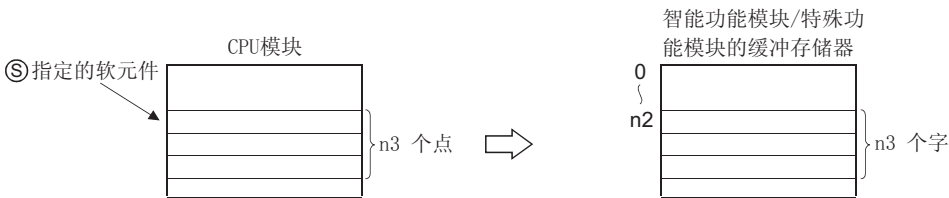


[设定数据]

设定数据	含义	数据类型
n1	智能功能模块/特殊功能模块的起始 I/O 号	BIN 16 位
n2	数据写入操作的起始地址	
Ⓢ	存储要写入数据的软元件的起始号	BIN 16/32 位
n3	写入数据的数量 ● T0(P) : 1 到 6144 ● DT0(P) : 1 到 3072 (仅适用于 A/QnA)	BIN 16 位

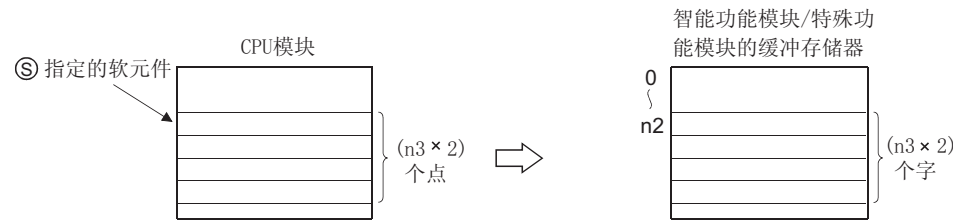
[功能]

- T0
- (1) 从Ⓢ指定的软元件上将 n3 个点的数据写入到 n1 指定的智能功能模块/特殊功能模块的缓冲存储器中从 n2 指定的地址开始的区域内。



DT0

- (1) 从⑤指定的软元件上将 $(n3 \times 2)$ 个点的数据写入到 $n1$ 指定的智能功能模块/特殊功能模块的缓冲存储器中从 $n2$ 指定的地址开始的区域内。
智能功能模块/特殊功能模块的缓冲存储器



要点

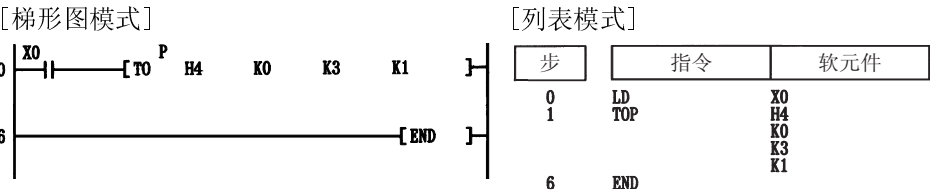
- (1) 智能功能模块/特殊功能模块的软元件也可用于从智能功能模块/特殊功能模块软元件的缓冲存储器上读取数据。
智能功能模块/特殊功能模块软元件的详细情况，请参考所使用的 CPU 模块的用户手册(功能解释, 编程基础)或 QnACPU 编程手册(基础)。

[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志(SM0)变为 ON，且错误代码存储在 SD0 上。
- 在执行指令时，没有同智能功能模块/特殊功能模块进行信号交换。(错误代码:1412)
 - 在执行指令时，智能功能模块/特殊功能模块上检测到错误。(错误代码:1402)
 - $n1$ 指定的 I/O 号不是用于智能功能模块/特殊功能模块。(错误代码:2110)
 - ⑤指定的软元件的 $n3$ (对于 DT0 是 $2 \times n3$ 个点) 个点超过了指定软元件的范围。(错误代码:4101)
 - $n2$ 指定的地址超出了缓冲存储器的范围。(错误代码:4100)
 - $n2$ 指定的地址是一个奇数地址(AJ71QC24(N))。(错误代码:4100)
 - 读取数据的数量超出了 6144/3072。(仅适用于 A/QnA 模块)(错误代码:4101)

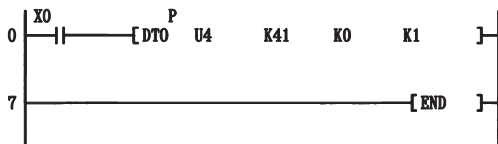
[程序示例]

- (1) 当 X0 变为 ON 时，以下程序将安装在 I/O 号 040 到 05F 的 A68AD 的 CH1 和 CH2 设置到“A/D 转换”(将“3”写入到缓冲存储器的地址 0)。



(2) 当 X0 变为 ON 时, 以下程序将安装在 I/O 号 040 到 05F 的 AD71 的 X 轴的当前值设置为 0。
(将 0 写入缓冲存储器的地址 41, 42)。

[梯形图模式]

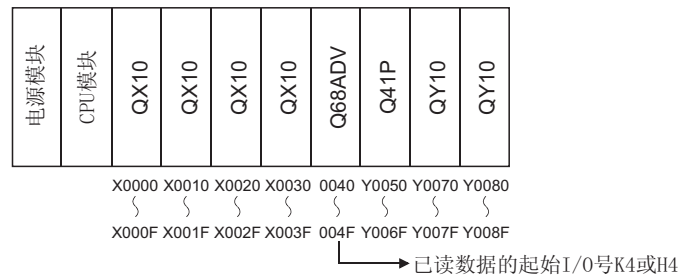


[列表模式]

步	指令	软元件
0	LD	X0
1	DTOP	U4 K41 K0 K1
7	END	

备注

1) 当以十六进制 4 位数表示安装智能功能模块/特殊功能模块的插槽的起始 I/O 号时, n1 的值由高三位指定。



2) QCPU 建立 T0/DT0 指令的自动互锁。

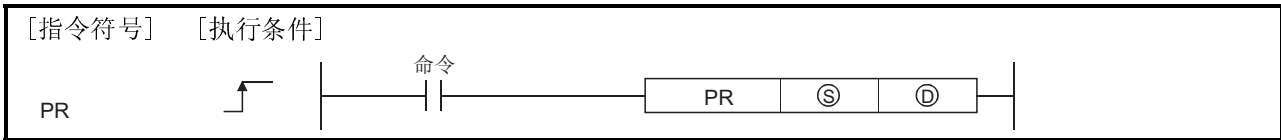
QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.9 显示指令

7.9.1 打印 ASCII 码指令 (PR)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 \$	其它
	位	字		位	字				
Ⓔ	—	△*		—			○	○	—
Ⓕ	○ (仅 Y)	—		—			○	—	—

※: 不能使用为每个局部软元件和程序设置的文件寄存器。

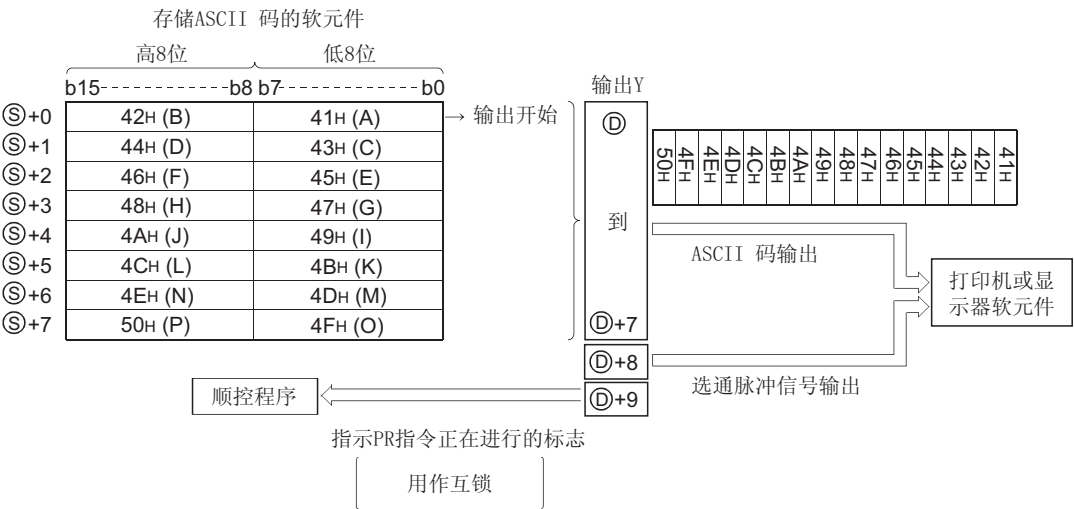


[设定数据]

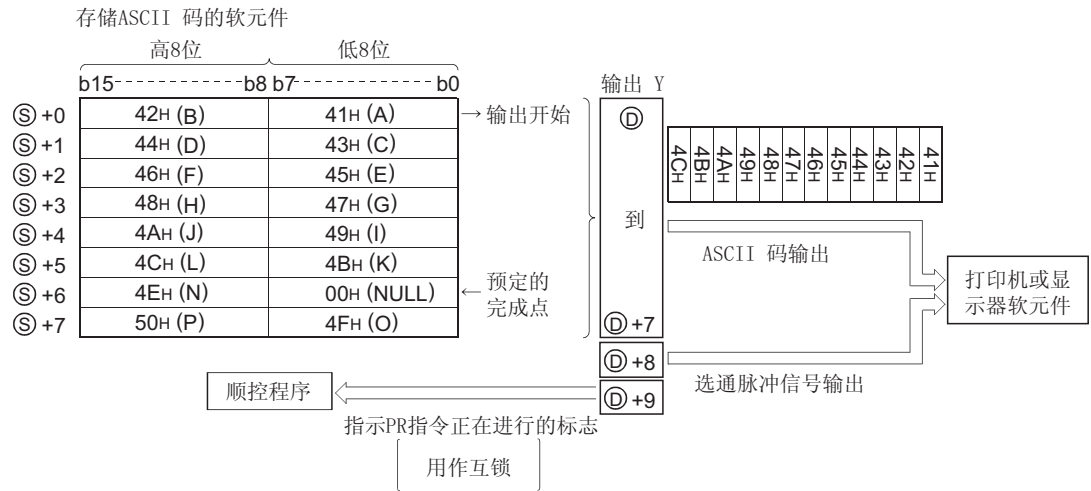
设定数据	含义	数据类型
	存储 ASCII 码的软元件的起始号	字符串
	输出 ASCII 码的输出模块的起始号	位

[功能]

- (1) 将存储在指定的软元件之后的 ASCII 码输出到指定的一个输出模块上。
输出字符数根据 SM701 的 ON/OFF 状态会有不同 (输出字符数选择)。
- (a) 如果 SM701 为 ON, 将对从指定的软元件开始的 8 个点的字符 (16 字符) 进行运算。



(b)如果 SM701 为 OFF，则从⑤指定软元件到 00H 码之间的的有数据都将作为操作焦点。

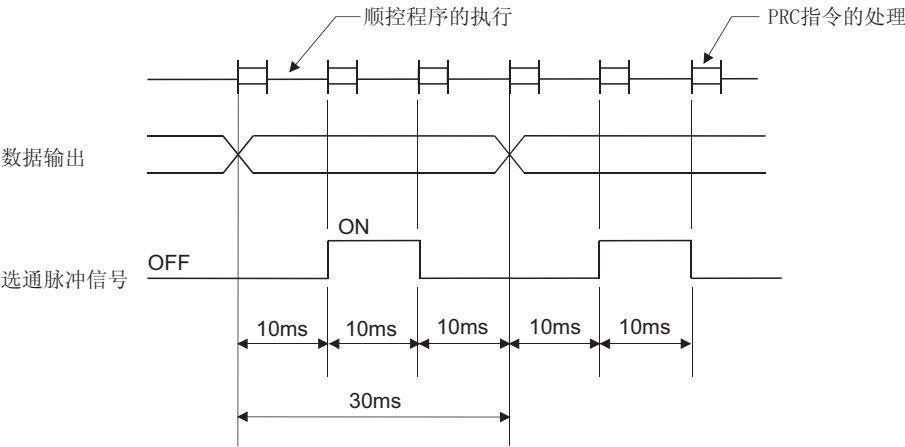


(2) 输出模块使用的点数是从⑩指定的 Y 地址开始的 10 点。

(3) 来自输出模块的输出信号以每 30ms 一个字符的速率进行传送。

因此，完成 (n) 个指定字符数的传送所需要的时间是 30ms×n (ms)。

每隔 10ms，PR 指令执行数据输出，选通脉冲信号为 ON，然后选通脉冲信号变为 OFF。在以上处理之间的时间，继续执行其它指令。



(4) 除了 ASCII 码之外，输出模块也从⑩+8 软元件上输出选通脉冲信号。

(5) 在执行 PR 指令之后，PR 指令执行标志 (⑩+9 软元件) 保持 ON 状态，直到完成指定字符数的传送。

(6) PR 和 PRC 指令可多次使用，但是最好用 PR 指令的执行标志 (⑩+9 软元件) 建立互锁，这样它们就不会同时处于 ON 状态。

(7) 如果在 ASCII 码的输出期间，存储 ASCII 码的软元件的内容被更改，则将输出更新的数据。

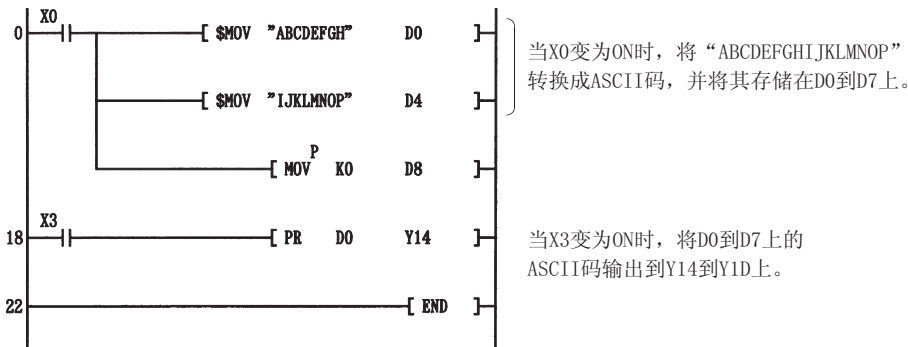
[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，且错误代码存储在 SD0 上。
- 当 SM701 为 OFF 时，在㊟指定的软元件范围内没有 00H 码。(错误代码:4101)

[程序示例]

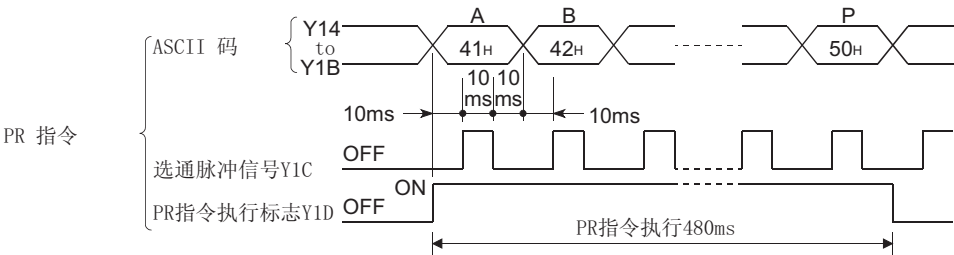
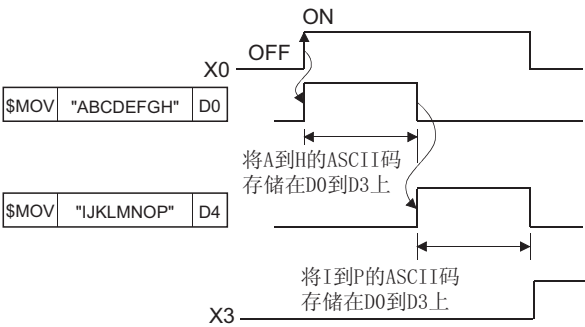
- (1) 当 X0 变为 ON 时，以下程序将字符串“ABCDEFGH IJKLMNOP”转换成 ASCII 码，并将其存储在 D0 到 D7 上，然后当 X3 变为 ON 时，将 D0 到 D7 上的 ASCII 码输出到 Y14 到 Y1D 上。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	\$MOV	"ABCDEFGH" D0
8	\$MOV	"IJKLMNOP" D4
15	MOVP	K0 D8
18	LD	X3
19	PR	D0 Y14
22	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.9.2 打印注释指令 (PRC)

(1) 使用高性能型 QCPU/过程 CPU 时

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [J][K]		变址寄存器 Zn	常量 \$
	位	字		位	字		
⑤	—	○		○		—	—
⑥	○ (仅 Y)	—		—		—	—

[指令符号]	[执行条件]
PRC	

[设定数据]

设定数据	含义	数据类型
⑤	要打印注释的软元件的起始号	软元件名称
⑥	要输出注释的输出模块的起始号	位

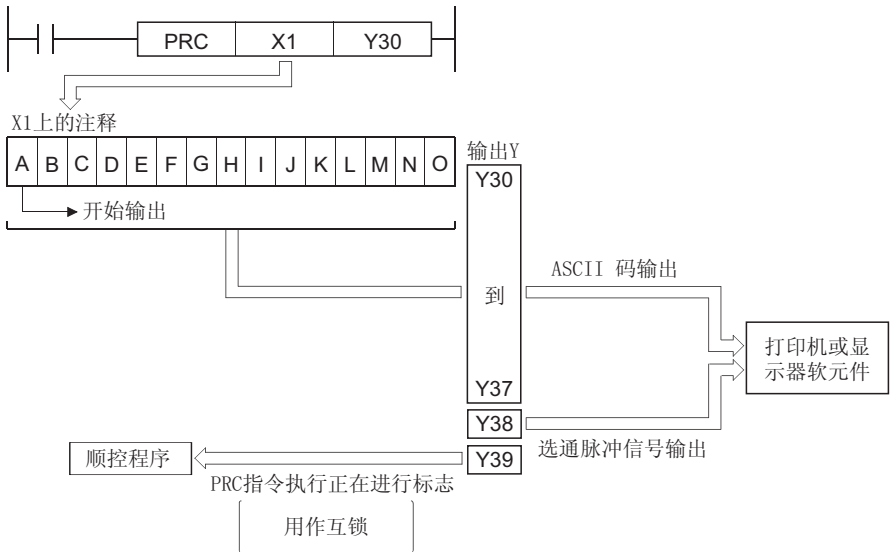
[功能]

(1) 将⑤指定的软元件上的注释 (ASCII 码) 输出到⑥指定的输出模块上。

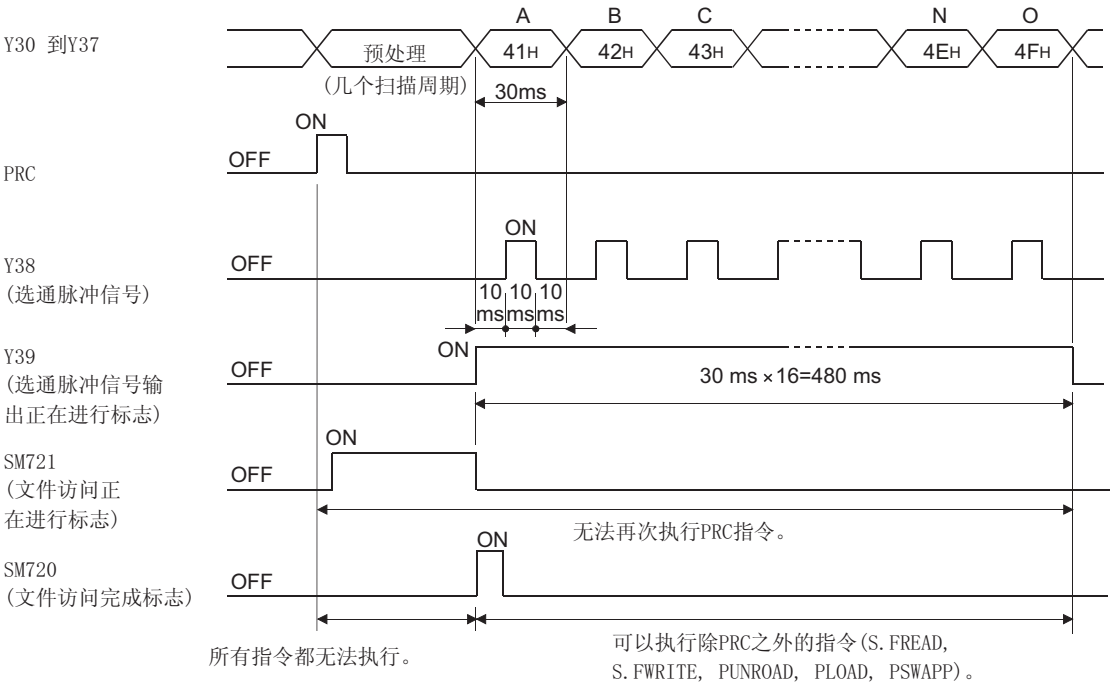
输出字符数将根据 SM701 的 ON/OFF 状态有所不同。

- 当 SM701 为 OFF 时：注释是 32 个字符
- 当 SM701 为 ON 时：注释是高 16 个字符

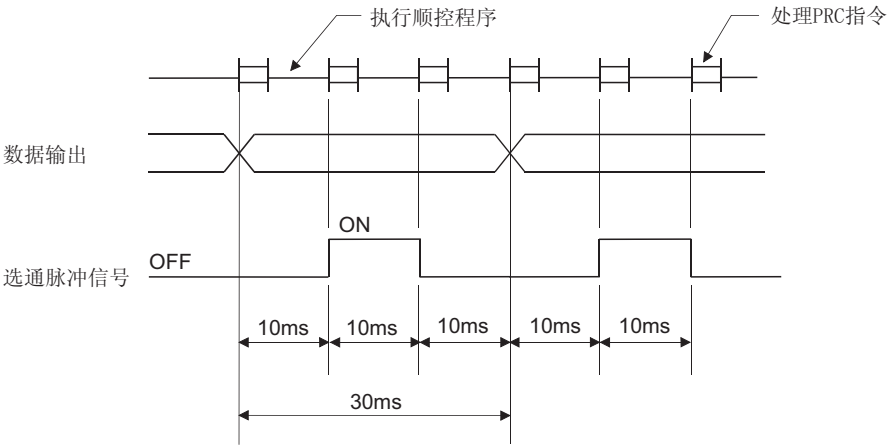
输出模块使用的点数是从⑥指定的 Y 地址开始的 10 个点。



[时序图]



- (2) 来自输出模块的输出信号以每 30ms 一个字符的速率进行传送。
因此，完成 (n) 个指定字符数的传送需要的时间是 $30 \times n$ (ms)。
每隔 10ms, PRC 指令执行数据输出, 选通脉冲信号为 ON, 然后选通脉冲信号为 OFF。在以上处理之间的时间, 继续执行其它指令。



- (3) 除了 ASCII 码之外, 输出模块也从⑤+8 软元件上输出选通脉冲信号 (10msON, 20msOFF)
- (4) 执行 PRC 指令之后, PRC 指令执行标志 (⑤+9 软元件) 保持 ON 状态, 直到完成指定字符数的传送。

- (5) PRC 指令可多次使用，但是最好用 PRC 指令执行标志 (D+9 软元件) 建立互锁，这样它们就不会同时处于 ON 状态。
- (6) 如果没有在⑤指定的软元件上登记注释，则不会进行处理。
- (7) 当注释被读取时，指令完成后 SM720 变为 ON 一个扫描周期。SM721 在指令执行期间变为 ON。当 SM721 为 ON 时，无法执行 PRC 指令，如果试图执行，则不会进行处理。

要点
(1) PRC 指令使用的软元件注释使用存放在存储卡上的注释文件。 它们不能使用内存中的注释文件。
(2) 在参数模式中的“PLC 文件设置”选项中设置 PRC 指令使用的注释文件。 如果未给 PLC 文件设置设定它所使用的注释文件，则无法使用 PRC 指令输出软元件注释。
(3) 不要在中断程序期间执行 PRC 指令。 否则，会导致故障出现。

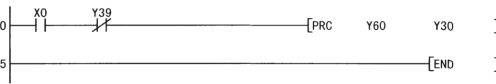
[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，且错误代码存储在 SD0 上。
- 在 RUN 期间当写入注释时执行 PRC 指令。(错误代码:4100)

[程序示例]

当 X0 变为 ON 时，程序将 Y60 的注释输出到 Y30 到 Y39 上。

[梯形图模式]

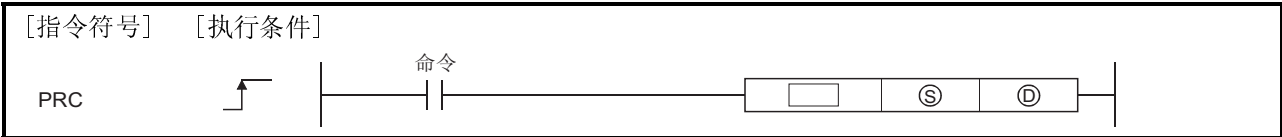


[列表模式]

步	指令	软元件
0	LD	X0
1	ANI	Y39
2	PRC	Y60
5	END	

(2) 使用 QnACPU 时

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 \$	其它
	位	字		位	字				P, I, J, U
⑤	○	○		○			—	—	○
⑥	○ (仅 Y)	—		—			—	—	—

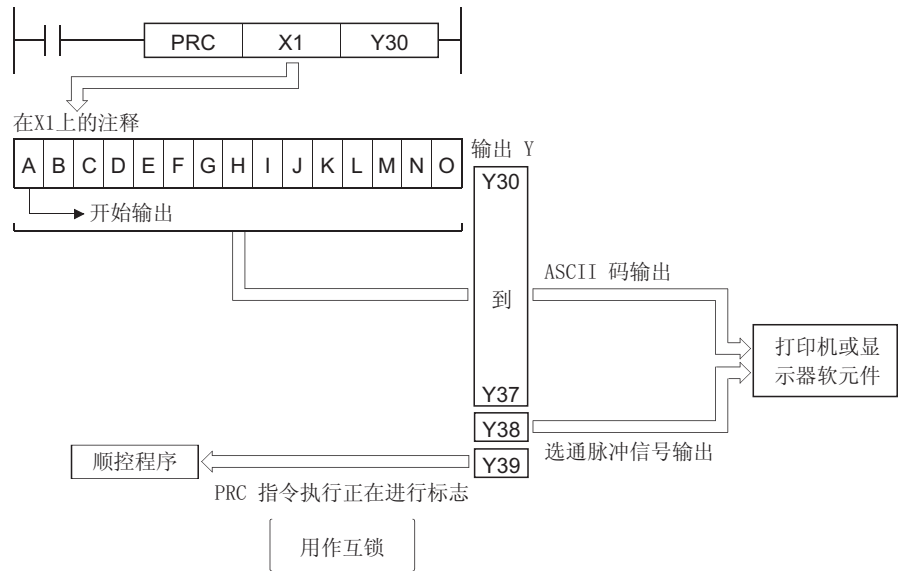


[设定数据]

设定数据	含义	数据类型
Ⓔ	打印注释的软元件的起始号	Devicename
Ⓕ	输出注释的输出模块的起始号	位

[功能]

- (1) 将Ⓔ指定的软元件上的注释 (ASCII 码) 输出到Ⓕ指定的输出模块上。
输出字符的数量将根据 SM701 的 ON/OFF 状态而有所不同。
- 当 SM701 为 OFF 时：注释是 32 字符
 - 当 SM701 为 ON 时：注释是高 16 字符
- 输出模块使用的点数是从Ⓕ指定的 Y 地址开始的 10 点。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	×	×	○	○

7.9.3 ASCII 码 LED 显示指令(LED)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [J][K][L]		特殊功能模 块 [U][V][G]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○		—			—

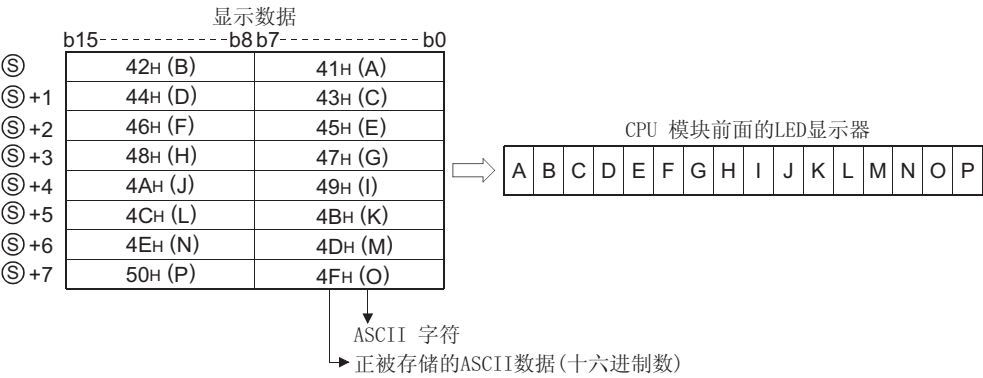


[设定数据]

设定数据	含义	数据类型
⑤	存储显示数据的软元件的起始号	字符串

[功能]

- (1) 在 CPU 模块前部的 LED 显示器上显示存储在从由⑤指定的软元件开始的 8 个数据点的 ASCII 数据 (16 字符)。



- (2) 如果从⑤指定的软元件开始的 8 个数据点上没有存储 ASCII 数据，则将发生以下状况：

- (a) T, C, D, W..... 空白
- (b) R..... 无法确定此处将显示什么 (如果文件寄存器(R)已清除，将是空白)

- (3) LED 显示器可显示以下字符：

- 数字0到9
- 字母 A 到 Z (大写字母)
- 特殊符号 <, >, =, *, /, ,, , +, -

(4) 在顺控程序中使用\$MOV 指令来转换包含字母和数字的 ASCII 数据。

[运行错误]

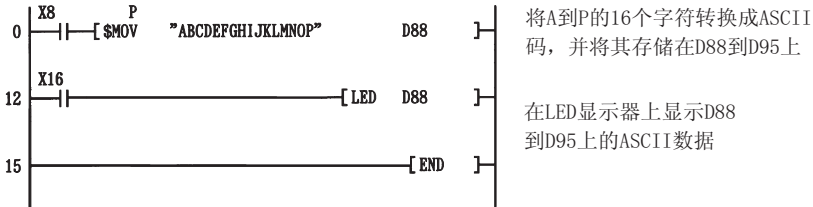
(1) 没有同 LED 指令相关的运行错误。

要点
该指令仅可用于 Q3ACPU, Q4ACPU 和 Q4ARCPU。 如果在一个前面没有 LED 显示器的 CPU 模块(QCPU, Q2ACPU, Q2AS (H) CPU) 上执行 LED 指令，则不会进行任何处理。

[程序示例]

(1) 当 X8 变为 ON 时，以下程序将字符串“ABCDEFGH IJKLMN OP”转换成 ASCII 码，并将其存储在 D88 到 D95 上，然后当 X16 变为 ON 时，将 D88 到 D95 上的 ASCII 码输出到位于 CPU 模块前面的 LED 显示器上。

[梯形图模式]



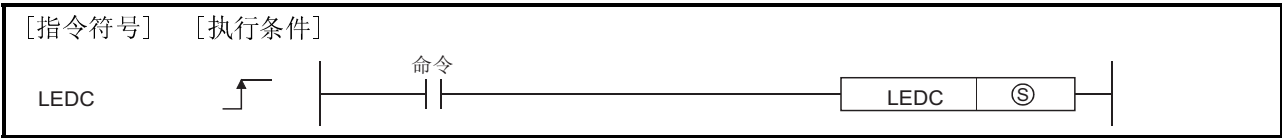
[列表模式]

步	指令	软元件
0	LD	X8
1	\$MOVP	"ABCDEFGH IJKLMN OP"
		D88
12	LD	X16
13	LED	D88
15	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	×	×	○	○

7. 9. 4LED 注释显示指令 (LEDC)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[] []		特殊功能模 块 U[] \ G[]	变址寄存器 Zn	常量 \$	其它
	位	字		位	字				BL, BL\S, BL\TR
⑤	○	○		○			—	—	○



[设定数据]

设定数据	含义	数据类型
⑤	用于显示注释的软元件的起始号	软元件名称

[功能]

- (1) 在位于 CPU 模块前面的 LED 显示器上显示存储在⑤指定的软元件上的注释(16 字符)。
- (2) 如果⑤指定的软元件上没有粘贴注释，或者注释超出了指定注释范围，则将发生以下状况：

⑤指定		LED 动作
在注释范围之内	粘贴注释	LED 显示器上显示注释
	未粘贴注释	清除 LED 显示器
超出注释范围		无动作(LED 显示不变)

- (3) 如果注释含有 LED 显示器无法显示的字符，则不会进行准确显示。
LED 显示器可以显示以下字符：
- 数字 0 到 9
 - 字母 A 到 Z(大写字母)
 - 特殊符号<, >, =, *, /, ,, +, -
- (4) LED 显示器上显示的注释最多可有 16 个字母，当软元件注释有 16 个或更多字母时，显示前 16 个字母。

[运行错误]

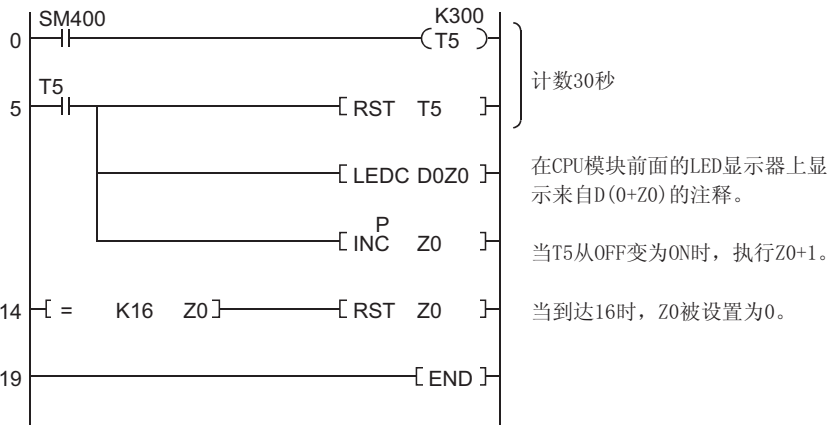
(1) LEDC 指令没有运行错误。

要点
该指令仅可用于 Q3ACPU、4ACPU 和 Q4ARCPU。 如果在一个前面没有 LED 显示器的 CPU 模块 (QCPU, Q2ACPU, Q2AS (H) CPU) 上执行, 则不会进行任何处理。

[程序示例]

(1) 以下程序在 30 秒的间隔内显示从 D0 到 D15 的注释。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	OUT	T5
		K300
5	LD	T5
6	RST	T5
10	LEDC	D0Z0
12	INCP	Z0
14	LD=	K16
		Z0
17	RST	Z0
19	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.9.5 错误显示和报警器复位指令 (LEDR)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G]	变址寄存器 Zn	常量	其它
	位	字		位	字				
—	—								



[功能]

将自诊断错误显示复位，以便报警器显示或操作能够继续。
通过执行一次该指令，错误显示或者报警器被复位。

- (1) 发生自诊断错误时的操作
- (a) 如果自诊断错误是一个允许继续操作的错误。

如果显示的自诊断错误是一个允许 CPU 模块继续操作的错误，则 CPU 前面的“ERROR/ERR.”LED 和错误显示将被复位。

必须在用户程序上复位 SM1 和 SD0，因为在此时它们不会被自动复位。

由于此时显示的错误的原因比报警器有较高的优越权，所以不采取任何复位报警器的措施。
- (b) 当发生电池错误时

如果在更换电池后执行 LEDR 指令，则 CPU 模块前面的“BAT.ARM/BAT.”LED 和错误显示将被复位。

此时 SM51 变为 OFF。
- (2) 当报警器 (F) 为 ON 时的操作
- (a) 当 CPU 模块没有 LED 显示器时

当执行 LEDR 指令时，将进行以下操作：

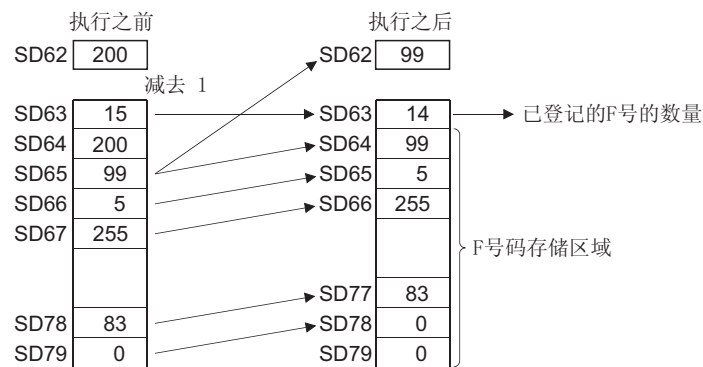
1) “USER” LED 闪烁，并被关掉。

2) 存储在 SD62 和 SD64 中的报警器 (F) 被复位，且代表 SD65 到 SD79 的 F 号被上移。

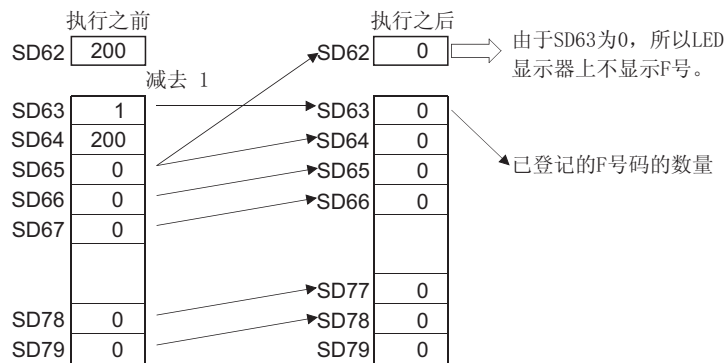
3) 新存储在 SD64 上的数据被传送到 SD62。

4) SD63 上的数据减 1。

但是，如果 SD63 为 0，则仍保持 0。



- (b) 对于前面带有 LED 显示器的 CPU
- 当执行 LEDR 指令时，将进行以下操作：
- 1) 显示在 CPU 模块前面的 F 号将被复位。
 - 2) “USER” 闪烁，并被关掉。
 - 3) 存储在 SD62 和 SD64 中的报警器 (F) 被复位，且代表 SD65 到 SD79 的 F 号被前移。
 - 4) 新存储在 SD64 上的数据被传送到 SD62。
 - 5) SD63 上的数据减 1。
- 但是，如果 SD63 为 0，则仍保持 0。
- 6) 在 LED 显示器上显示存储在 SD62 上的 F 号。
- 但是，如果 SD63 值为 0，则不显示。



备注

(1) 特殊寄存器 SD207 到 SD209 中设置的错误条目号和优先级的默认值如下表所示：

优先级	错误条目号 (十六进制)	说明	备注
1	1	ACDOWN	断电
2	2	UNITVERIFYERR. FUSEBREAKOFF P. UNITERROR	I/O 模块校验错误 保险丝烧坏 特殊功能模块校验错误
3	3	OPERATINERROR SFCPOPE. ERROR SFCPEXE. ERROR	运行错误 SFC 指令运行错误 SFC 程序执行错误
4	4	ICM. OPEERROR FILEOPE. ERROR EXTENDINST. ERROR	存储卡运行错误 文件访问错误 扩展指令错误
5	5	PRG. TIMEOVER	固定扫描设置超时错误 低速执行监视超时错误
6	6	CHK 指令	
7	7	报警器	
8	8	LED 指令	
9	9	BATTERYERR.	
10	A	时钟数据	

(2) 如果报警器被给予了最高优先权，可通过 LEDR 指令优先将其复位。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.10 调试和故障诊断指令

7.10.1 特殊格式故障检测 (CHKST, CHK)

设定数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G]	变址寄存器 Zn	常量	其它
	位	字		位	字				
—	—								

[指令符号] [执行条件]

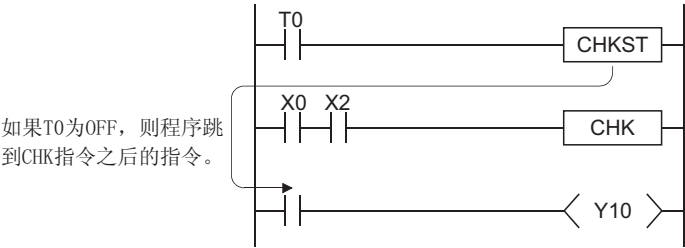
CHKST

CHK

[功能]

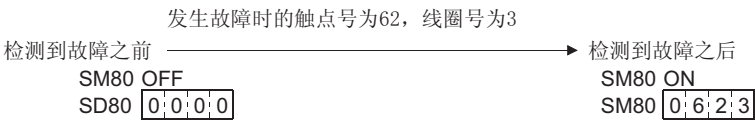
CHKST

- (1) CHKST 指令是启动 CHK 指令的指令。
- 如果 CHKST 指令的命令为 OFF，则执行跳到 CHK 指令之后的步骤。
- 如果 CHKST 指令的命令为 ON，则执行 CHK 指令。

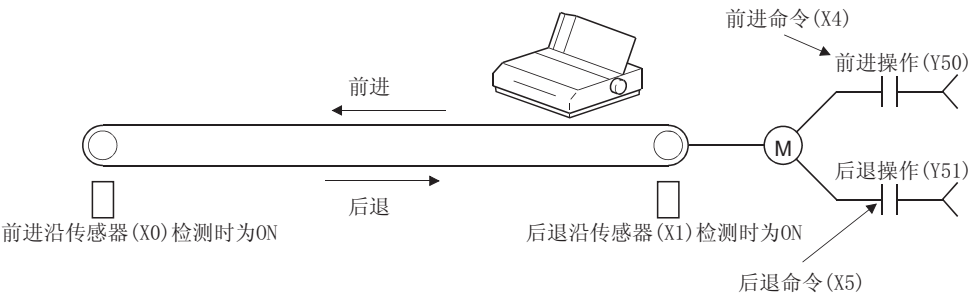


CHK

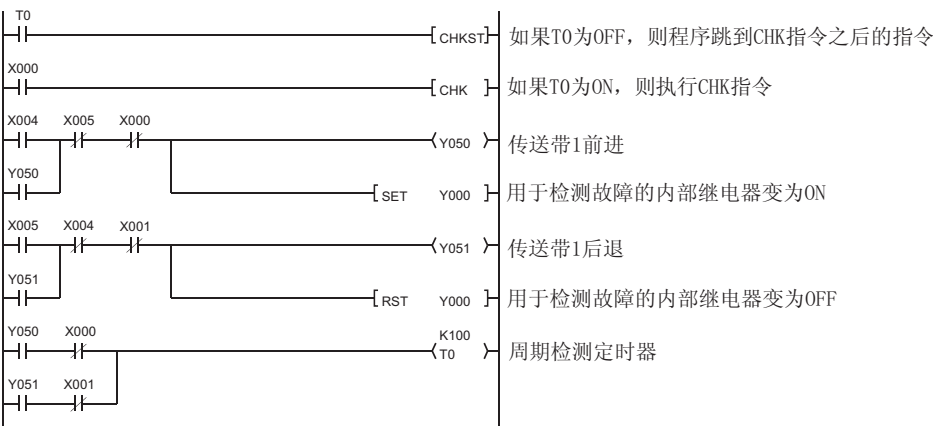
- (1) CHK 指令是用于下页所示的双向操作的指令，以确认系统故障的性质。
- (a) 当执行 CHK 指令时，通过指定的检测条件进行故障诊断检测，如果检测出故障，则 SM80 变为 ON，且以 BCD 值将故障号存储到 SD80 上。
- 发现故障的触点号存储在 SD80 的高 3 位数字中(参见(3))，发生故障的线圈号(参见(2))存储在 SD80 的低位数字上。



(b) CHK 指令之前的触点指令不控制 CHK 指令的执行，但却设置检测条件。



(c) 可以建立如下所示的梯形图，对上面的系统进行一个周期的检测：



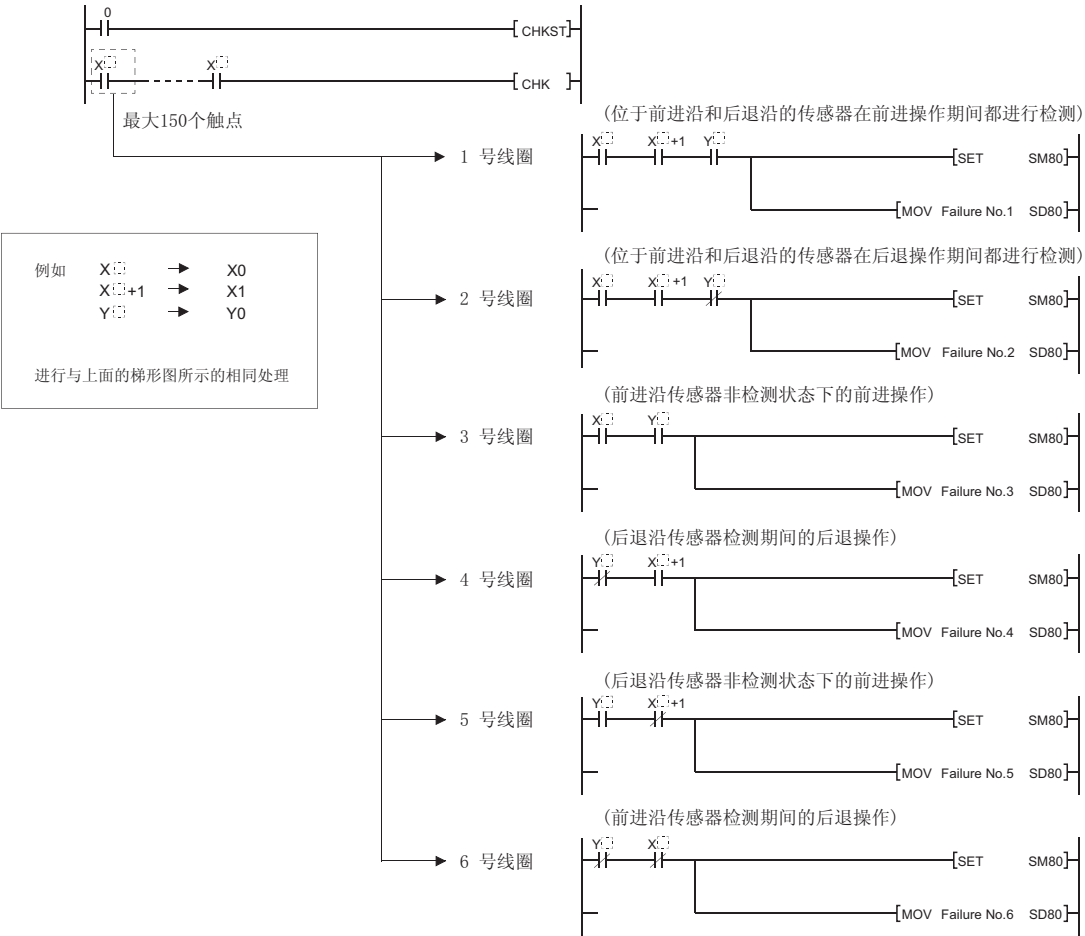
(d) 当创建使用 CHK 指令的梯形图时，应考虑以下几点：

- 1) 前进沿检测传感器和后退沿检测传感器的触点号 (X_i) 必须始终保持连续，此外，前进沿检测传感器的触点号 (X_i) 应该低于后退沿检测传感器的触点号。
- 2) 对前进沿检测传感器触点号 (X_i) 的控制和用相同号码 (Y_i) * 的输出如下：
当正在进行前进操作时 变为 ON
当正在进行后退操作时 变 OFF

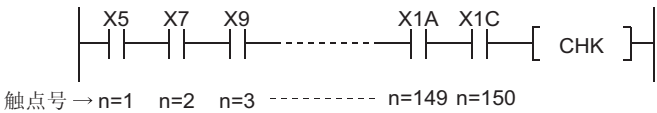
备注

1) *: 输出 (Y_i) 被视为一个内部继电器，不能输出到一个外部软元件上。

(2) 依据指定的触点，CHK 指令执行与以下梯形图所表示的处理相同的处理。



(3) 位于左边的垂直总线上的数字 1 到 150 已经在故障检测期间被分配为触点号。



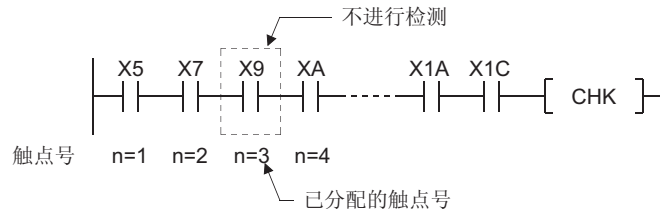
- (4) 在强制执行 CHK 指令之前复位 SM80 和 SD80。
在执行 CHK 指令之后，直到 SM80 和 SD80 已经被复位，才可以再次执行 CHK 指令。
(保留 SM80 和 SD80 的内容，直到用户复位)。
- (5) 必须在 CHK 指令之前放置一个 CHKST 指令。
- (6) CHK 指令可在程序的任何步被写入。
在所有正被执行的程序文件中最多可在两个位置使用 CHK 指令。
然而，CHK 指令无法在一个程序文件的 2 个以上的位置使用。

- (7) 在 CHK 指令之前放置 LD 和 AND 指令，以建立检测条件。

检测条件无法使用其它触点指令设置。

如果已经使用 LDI 或 ANI 设置了检测条件，则不会对它们指定的检测条件进行处理。

但是，故障检测期间的触点号也可以分配给 LDI 和 ANI 指令。



- (8) 故障检测方法将依据 SM710 为 ON 状态还是 OFF 状态而有所不同。

- (a) 如果 SM710 为 OFF，则逐次地对每个触点的 1 到 6 号线圈进行检测。

当执行 CHK 指令时，检测的顺序是从 1 号触点的 1 号线圈开始到 6 号线圈，然后转移到 2 号触点，从 1 号线圈开始检测。

当 n 号触点的 6 号线圈已经被检测时，CHK 指令将结束。

- (b) 如果 SM710 为 ON，检测将按照线圈号顺序从 1 号触点到 n 号触点进行。

当执行 CHK 指令时，检测将从 1 号线圈的梯形图开始，按照从 1 号触点到 n 号触点的顺序，然后转移到 2 号线圈的梯形图，从 1 号触点开始。

当 6 号线圈的 n 号触点已经被检测时，CHK 指令将结束。

- (9) 如果检测出一个以上的故障，第一个检测出的故障号将被存储。

此后检测的故障号将被忽略。

- (10) CHK 指令不可以被低速执行类型程序使用。

如果已经在一个包含 CHK 指令的程序文件中设置了一个低速执行类型程序，将返回一个运行错误，并且将暂停 CPU 模块的操作。

[运行错误]

- (1) 在发生以下运行错误的情况中，错误标志 (SM0) 变为 ON，且错误代码存储在 SD0 上。

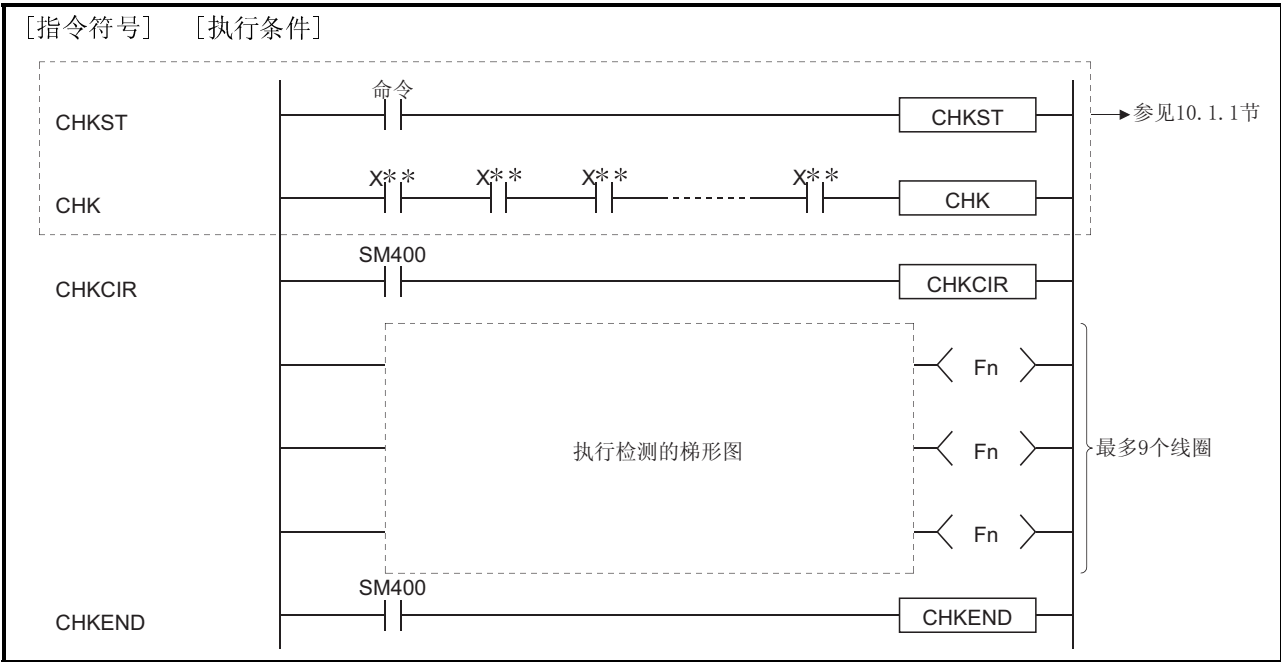
- 有一个平行的梯形图。 (错误代码:4235)
- 有一个 NOP 指令。 (错误代码:4235)
- 有 150 个以上的触点指令。 (错误代码:4235)
- 在 CHKST 指令之后未执行 CHK 指令。 (错误代码:4235)
- 未执行 CHKST 指令就执行 CHK 指令。 (错误代码:4235)
- 在一个低速执行类型程序中使用 CHKST 和 CHK 指令。 (错误代码:4235)

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.10.2 改变 CHK 指令的检测格式 (CHKCIR, CHKEND)

(1) 当使用 GX Developer 时 (高性能型 QCPU/过程控制 CPU/QnACPU)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		变址寄存器 Zn	其它
	位	字		位	字		
—	—						



[功能]

CHKCIR, CHKEND

- (1) CHK 指令中使用的检测梯形图的样式可更新到任何需要的格式。
实际的故障检测是通过 CHKST 和 CHK 指令执行的。
- (2) 根据 CHK 指令指定的检测条件以及 CHKCIR 和 CHKEND 指令之间的梯形图执行故障检测。

备注

有关 CHKST 和 CHK 指令的更多信息，请参见 7.10.1 节。

要点

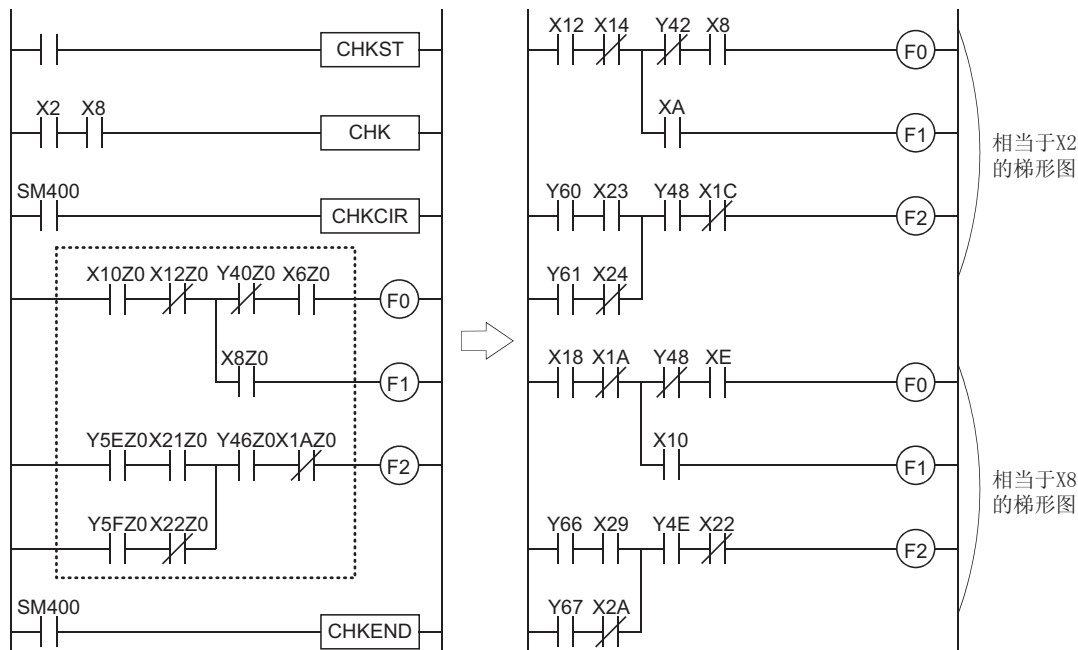
- (1) 为了使用 CHKCIR 到 CHKEND 指令来改变 CHK 指令的检测格式，用户应该通过变址修改 (Z0) 来创建一个梯形图。

(a) 检测条件(下图中的 X2 和 X8)处指示的软元件号将被假设为用于梯形图的单个软元件号(报警器(F)除外)的变址修改值。

示例：下图中□中的 X10 如下所示：
当对应检测条件 X2 时 由 X12 执行处理
当对应检测条件 X8 时 由 X18 执行处理

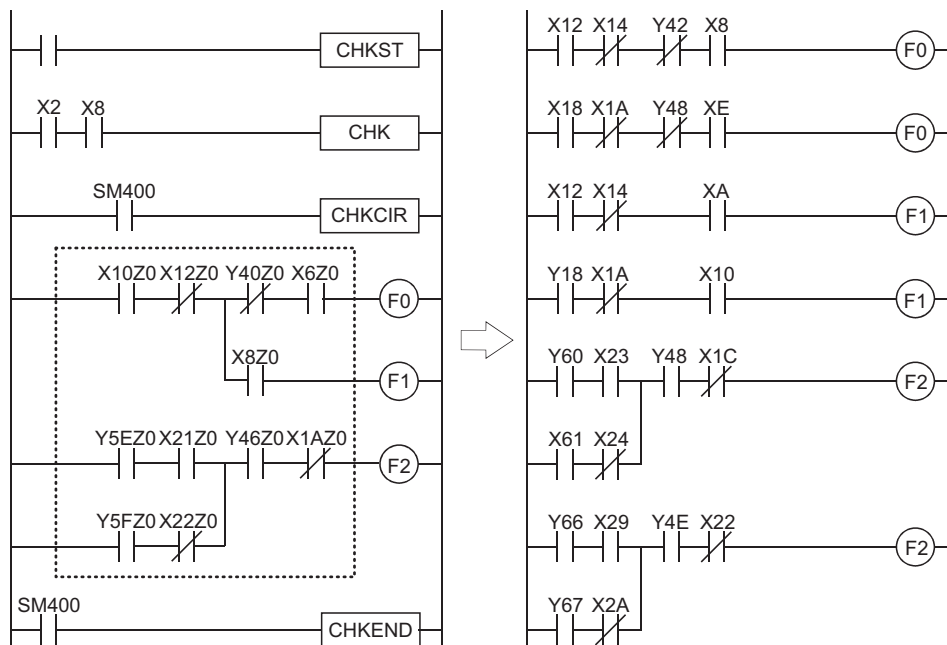
但是，根据 SM710 为 ON 状态还是 OFF 状态，故障检测执行的顺序会有所不同。
1) 如果 SM710 为 OFF，检测将依次对每个触点从 1 号线圈开始，执行到最后。

[CHKCIR 到 CHKEND 指定的梯形图] [CPU 模块检测顺序]



2) 如果 SM710 为 ON, 检测将按照线圈号顺序, 从 1 号触点执行到最后。

[CHKCIR 到 CHKEND 指定的梯形图] [CPU 模块检测顺序]



(b) 故障检测通过使用各种检测条件下的梯形图来检测 OUT F_□ 的 ON/OFF 状态。

在所有检测条件中, SM80 将变为 ON, 即使 OUT F_□ 中只有一个在梯形图模式中变为 ON 状态。

此外, 与被发现处于 ON 状态的 OUT F_□ 相对应的错误号 (触点号和线圈号), 将被从 SD80 开始按照 BCD 顺序存储起来。

(c) 梯形图中可用的指令如下所示:

触点 ... LD, LDI, AND, ANI, OR, ORI, ANB, ORB, MPS, MPP, MRD 和比较运算指令

线圈 ... OUT F_□

(d) 以下软元件可用于梯形图触点:

输入 (X), 输出 (Y)

(e) 仅报警器 (F) 可用于梯形图线圈中。

但是, 由于报警器 (F) 被用作虚拟触点, 所以它们可设置在 F0 和任意值之间。

此外, 它们可毫无困难地被重叠使用。

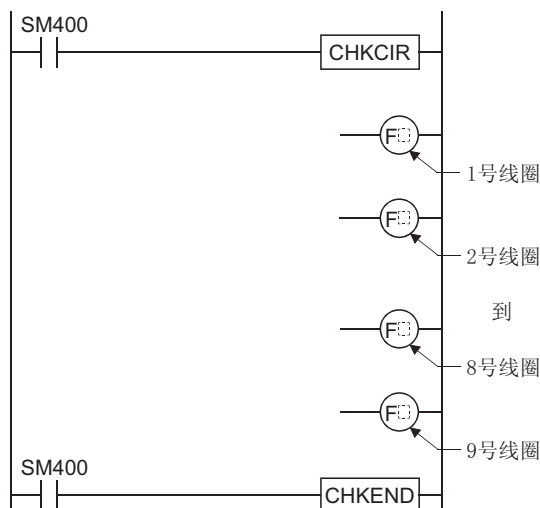
(f) 如果 CHK 指令执行期间使用的报警器 (F) 和某个其它非 CHK 指令环境中使用的报警器 (F) 有相同的号码, 仍可执行 ON/OFF 控制, 而不会产生错误。

在 CHK 指令执行期间, 对它们的处理是不同的, 而不是将它们看作处于不同语境中。

(g) 由于 CHK 指令中使用的报警器 (F) 实际上并没有变为 ON/OFF, 如果被一个外围设备监视, 它们将不会变为 ON/OFF。

- (h) 最多可用 256 步创建一个梯形图。
此外，OUT F 最多可用 9 个线圈。

(3) 用于被 CHKCIR 到 CHKEND 指令指定的梯形图的线圈号，从 1 到 9，从头到尾，依次分配。



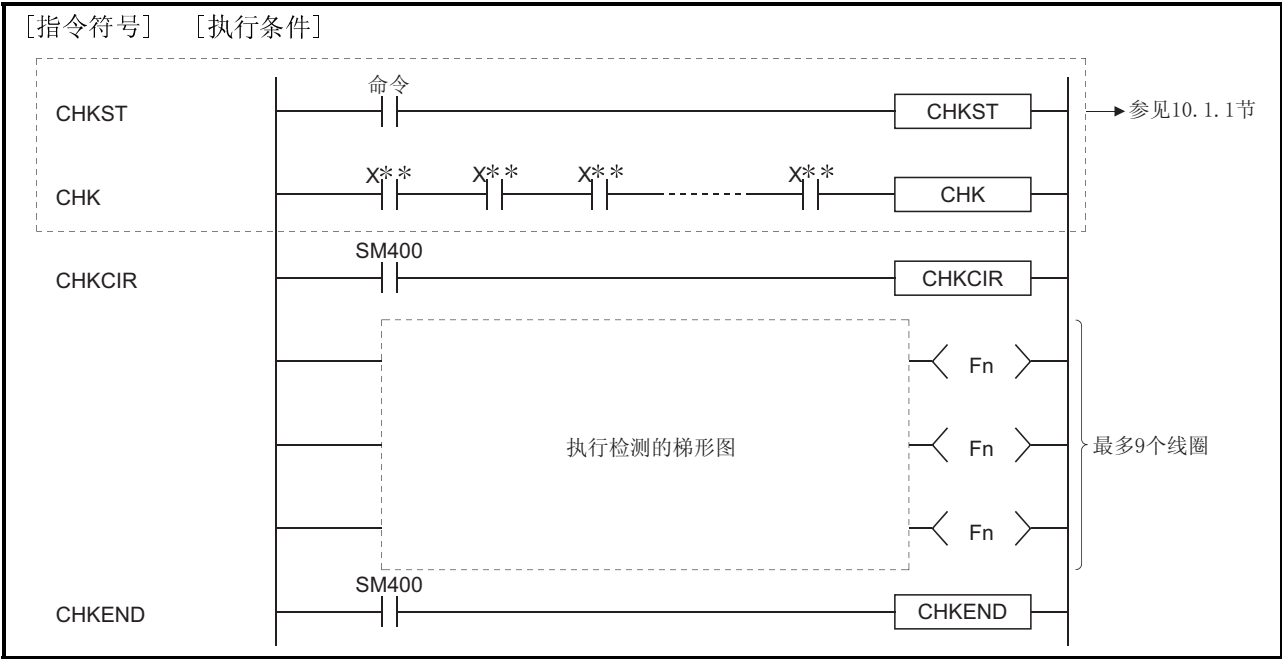
- (4) CHKCIR 和 CHKEND 指令可在程序中任何需要的步被写入。
在所有正被执行的程序文件中，最多可在两个位置使用 CHKCIR 和 CHKEND 指令。
但是，CHKCIR 和 CHKEND 指令在单个程序文件中不能在 1 个以上的位置使用。
- (5) CHKCIR 和 CHKEND 指令不能在低速执行类型程序中使用。
如果一个写有 CHKCIR 或 CHKEND 指令的程序文件被设置成一个低速执行类型程序，则将会发生运行错误，高性能型 QCPU/过程控制 CPU/QnACPU 的操作将被暂停。

[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 上。
- CHKCIR 或 CHKEND 指令在所有程序文件中出现三次或更多次。 (错误代码: 4235)
 - CHKCIR 或 CHKEND 指令在单个程序文件中出现两次或更多次。 (错误代码: 4235)
 - CHKCIR 指令执行之后未执行 CHKEND 指令。 (错误代码: 4230)
 - 在未执行 CHKCIR 指令的情况下执行 CHKEND 指令。 (错误代码: 4230)
 - CHKST 和 CHK 指令用于低速类型程序。 (错误代码: 4235)
 - 在一个梯形图中有 10 个或更多个 F 实例。 (错误代码: 4235)
 - 梯形图有 257 个或更多的步。 (错误代码: 4235)
 - 在梯形图中碰到无法使用的软元件。 (错误代码: 4235)
 - 已经在一个梯形图软元件上执行了变址修改。 (错误代码: 4235)

(2) 当使用 GPPQ 时 (QnACPU)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]G[]	变址寄存器 Zn	常量	其它
	位	字		位	字				
—	—								



[功能]

CHKCIR, CHKEND

- (1) 在 CHK 指令中将使用的检测梯形图可以更新到任何需要的格式。
实际故障检测通过 CHKST 和 CHK 指令执行。
- (2) 根据 CHK 指令指定的检测条件以及 CHKCIR 和 CHKEND 指令之间的梯形图执行故障检测。

备注

有关 CHKST 和 CHK 指令的更多信息，请参见 7. 10. 1 节。

要点

当使用 CHKCIR 到 CHKEND 指令并更改 CHK 指令的检测格式时，需要在“一般模式”中启动 GPPQ，以创建程序，并执行程序扩展。
如果已经通过一个 CPU 类型名称 (Q2A, Q2A-S1, Q2AS, Q2AS-S1, Q2ASH, Q2ASH-S1, Q3A, Q4A, Q4AR) 去启动 GPPQ，然后试图使用 CHKCIR 到 CHKEND 指令为 CHK 指令创建一个检测格式更新梯形图，则将无法执行准确处理。

(a) 检测条件(下图中的 X2 和 X8)上指示的软元件号将被假定为梯形图中的单个软元件号(报警器(F)除外)的变址修改值。

示例: 下图中[]中的 X10 如下所示:

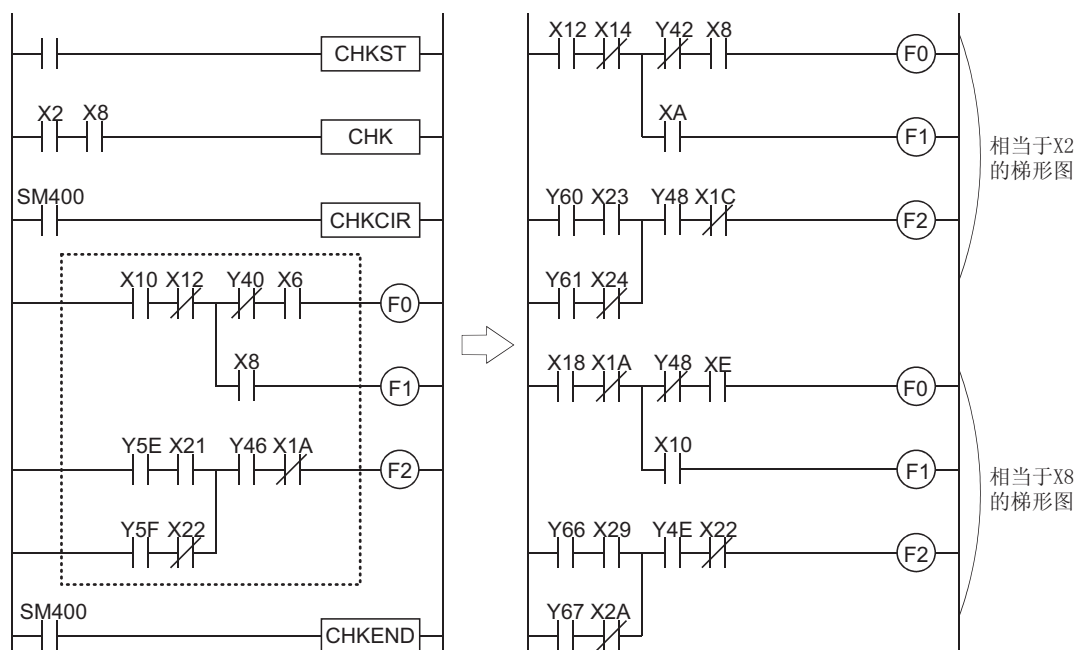
当对应检测条件 X2 时 由 X12 执行处理

当对应检测条件 X8 时 由 X18 执行处理

但是, 根据 SM710 为 ON 状态还是 OFF 状态, 故障检测执行的顺序会有所不同。

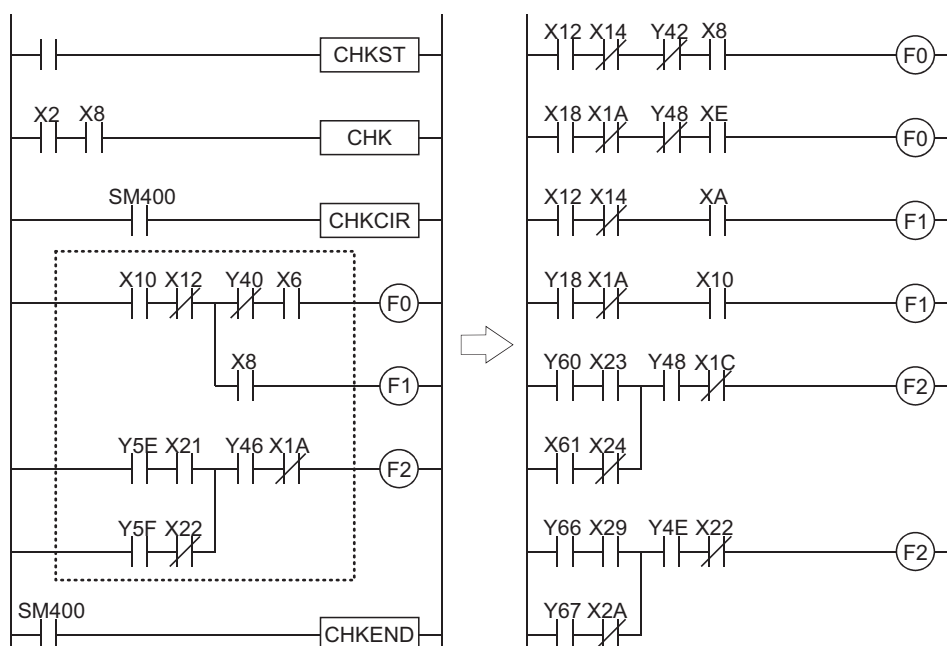
1) 如果 SM710 为 OFF, 检测将依次对每个触点从 1 号线圈开始, 执行到最后。

[CHKCIR 到 CHKEND 指定的梯形图] [QnACPU 检测顺序]



2) 如果 SM710 为 ON, 检测将按照线圈号顺序从 1 号触点开始, 执行到最后。

[CHKCIR 到 CHKEED 指定的梯形图] [QnACPU 检测顺序]



(b) 故障检测通过使用各种检测条件下的梯形图来检测 OUT F₀ 的 ON/OFF 状态。

在所有检测条件中, SM80 将变为 ON, 即使 OUT F₀ 中只有一个在梯形图中处于 ON 状态。

此外, 与被发现处于 ON 状态的 OUT F₀ 相对应的错误号 (触点号和线圈号), 将被从 SD80 开始按照 BCD 顺序存储起来。

(c) 梯形图中可用的指令如下:

触点 ... LD, LDI, AND, ANI, OR, ORI, ANB, ORB, MPS, MPP, MRD 和比较运算指令

线圈 ... OUT F₀

(d) 以下软元件可用于梯形图触点:

输入 (X), 输出 (Y)

(e) 仅报警器 (F) 可用于梯形图线圈中。

但是, 由于报警器 (F) 被用作虚拟触点, 所以它们可设置在 F0 和任意值之间。

此外, 它们可毫无困难地重叠。

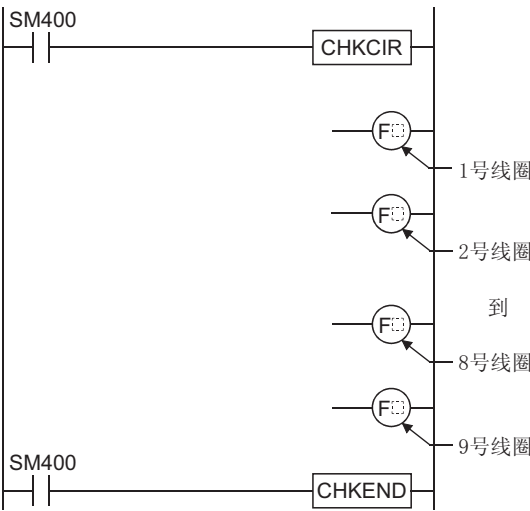
(f) 如果在 CHK 指令执行期间使用的报警器 (F) 和某个其它非 CHK 指令环境中使用的报警器 (F) 有相同的号码, 则仍可执行 ON/OFF 控制, 而不会产生错误。

在 CHK 指令执行期间, 对它们的处理是不同的, 而不是将它们看作处于不同语境中。

(g) 由于 CHK 指令中使用的报警器实际上并没有变为 ON/OFF, 如果受 GPPQ 监视, 它们将不会变为 ON/OFF。

(h) 最多可用 256 步创建一个梯形图。
此外，OUT F 最多可用 9 个线圈。

(3) 用于被 CHKCIR 到 CHKEND 指令指定的梯形图的线圈号，被从 1 到 9，从头到尾，依次分配。



(4) CHKCIR 和 CHKEND 指令可在程序中任何需要的步被写入。
在所有正被执行的程序文件中，最多可在两个位置使用 CHKCIR 和 CHKEND 指令。
但是，CHKCIR 和 CHKEND 指令在单个程序文件中不能在 1 个以上的位置使用。

(5) CHKCIR 和 CHKEND 指令不能在低速执行类型程序中使用。
如果一个写有 CHKCIR 或 CHKEND 指令的程序文件被设置成一个低速执行类型程序，则将发生运行错误，QnACPU 操作将被暂停。

[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 上。
- CHKCIR 或 CHKEND 指令在所有程序文件中出现三次或更多次。(错误代码: 4235)
 - CHKCIR 或 CHKEND 指令在单个程序文件中出现两次或更多次。(错误代码: 4235)
 - CHKCIR 指令执行之后未执行 CHKEND 指令。(错误代码: 4230)
 - 在未执行 CHKCIR 指令的情况下执行 CHKEND 指令。(错误代码: 4230)
 - CHKST 和 CHK 指令用于低速类型程序。(错误代码: 4235)
 - 在一个梯形图中有 10 个或更多个 F 实例。(错误代码: 4235)
 - 梯形图有 257 个或更多的步。(错误代码: 4235)
 - 在梯形图中碰到不能使用的软元件。(错误代码: 4235)
 - 已经在在一个梯形图软元件上执行了变址修改。(错误代码: 4235)

要点
如果在一个 GPPQ 上，在程序扩展期间发生以下错误，可能使得扩展无法执行： <ul style="list-style-type: none">● 在梯形图中碰到不能使用的软元件。● 已经在在一个梯形图软元件上执行了变址修改。 如果发生任何以上错误，则要改正梯形图。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	×	×	○	○

7.10.3 设置和复位状态锁存(SLT, SLTR)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G[]	变址寄存器 Zn	常量	其它
	位	字		位	字				
—	—								



[功能]

SLT

- (1) 当执行 SLT 指令时，在 SLT 指令执行时刻，设置在编程工具的设置软元件上的数据被存储在存储卡的文件中，供状态锁存使用。
- (2) 仅当 SM806 为 ON 时，SLT 指令才有效。
如果 SM806 已经变为 OFF，则可以执行 SLT 指令，但是不会加以处理，并且不会执行状态锁存。
- (3) 当执行 SLT 指令时 SM807 变为 ON，当完成状态锁存时，SM808 变为 ON。
另外，执行状态锁存的步数被存储在 SD807 上。
- (4) SLT 指令已经被执行一次，将在第二次和之后试图执行时被忽略。
可以通过执行 SLTR 指令重新允许执行 SLT 指令。
- (5) 可以在编程工具上监视状态锁存操作的结果。

SLTR

- (1) SLTR 指令用于复位 SLT 指令。
- (2) 执行 SLTR 指令允许重新执行 SLT 指令。

备注

- 1) 有关状态锁存的详细资料，请参阅使用的 CPU 模块的用户手册。
- 2) 有关编程工具上状态锁存的执行和监视结果，请参阅编程工具的操作手册。

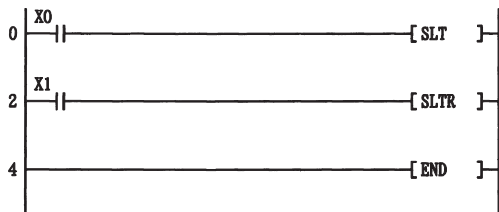
[运行错误]

(1) 没有和 SLT 或 SLTR 指令相关的运行错误。

[程序示例]

(1) 当 X0 变为 ON 时，以下程序执行 SLT 指令，当 X1 变为 ON 时，通过 SLTR 指令将其复位。

[梯形图模式]



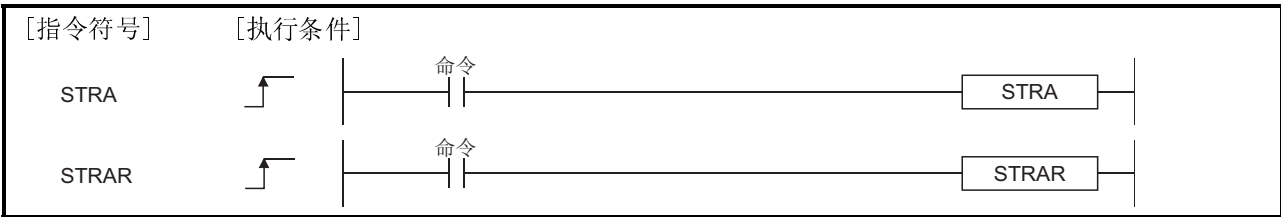
[列表模式]

步	指令	软元件
0	LD	
1	SLT	X0
2	LD	
3	SLTR	X1
4	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	×	×	○	○

7.10.4 设置和复位采样跟踪 (STRA, STRAR)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]G[][]	变址寄存器 Zn	常量	其它
	位	字		位	字				
—	—								



[功能]

STRA

- (1) 采样跟踪就是，在 SM800、SM801 和 SM802 变为 ON 时，将用编程工具设置的采样跟踪数据以预设的次数存储到存储卡的采样跟踪文件上。
当执行 STRA 指令时，SM803 变为 ON，并且在执行 STRA 指令后，采样已经执行了采样跟踪周期的设定数之后，数据被检测，并暂停采样跟踪操作。
- (2) 如果 SM801 在执行采样跟踪期间变为 OFF，则暂停采样。
- (3) 在 STRA 指令结束之后，完成采样跟踪，SM805 变为 ON。
- (4) 在已经执行了一次 STRA 指令之后，将在第二次和之后出现 STRA 指令时将其忽略。
运行 STRAR 指令，以使得 STRA 指令再次有效。

STRAR

- (1) STRAR 指令用于复位 STRA 指令。
- (2) 运行 STRAR 指令将使得 SM803 到 SM805 变为 OFF。
- (3) 运行 STRAR 指令，以使得 STRA 指令再次有效。

备注

- (1) 有关采样跟踪操作的详细资料，请参阅使用的 CPU 模块的用户手册。
- (2) 有关在编程工具上执行采样跟踪的情况，请参阅编程工具的操作手册。

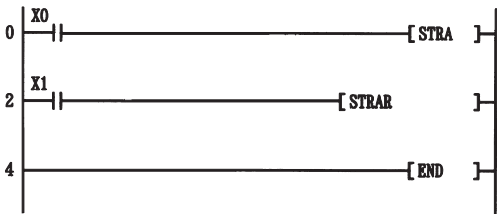
[运行错误]

- (1) 没有和 STRA 指令相关的运行错误。

[程序示例]

- (1) 当 X0 变为 ON 时，以下程序示例运行 STRA 指令，并且当 X1 变为 ON 时，使用 STRAR 指令复位 STRA 指令。

[梯形图模式]



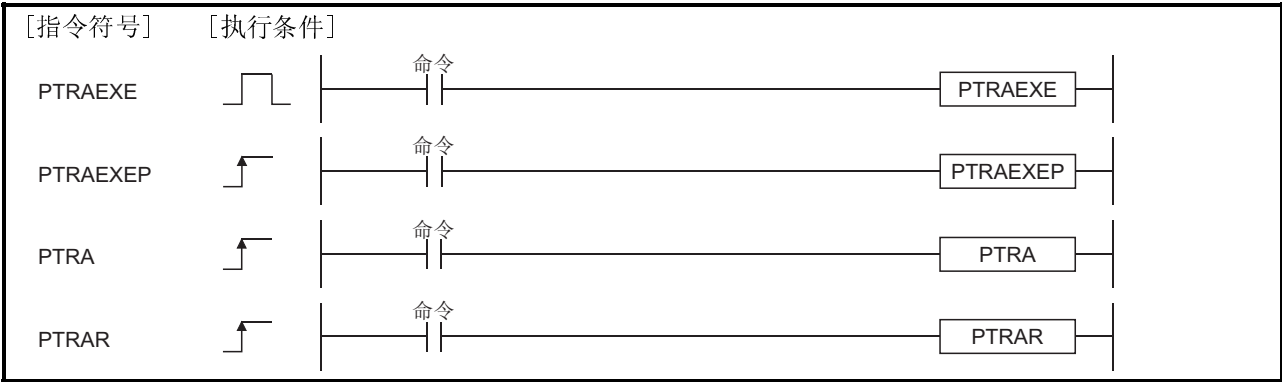
[列表模式]

步	指令	软元件
0	LD	X0
1	STRA	
2	LD	X1
3	STRAR	
4	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	×	×	○	○

7.10.5 程序跟踪的执行、置位和复位
(PTRAEXE, PTRAEXEP, PTRAR, PTRAR)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常量	其它
	位	字		位	字				
—	—								



[功能]

PTRA

- (1) 程序跟踪包括，将编程工具上设置的程序跟踪数据以 SM810, SM811 和 SM812 变为 ON 时设定的次数存储到 IC 存储卡上的程序跟踪文件中。
- (2) 当执行 PTRA 指令时，SM813 变为 ON，在采样已经执行了为 PTRA 指令执行操作之后设定的程序跟踪次数后，数据被锁存，程序跟踪操作被暂停。
- (3) 如果 SM811 在执行程序跟踪期间变为 OFF，则暂停采样。
- (4) 在执行 PTRA 指令之后，并且在程序跟踪结束之时，SM815 将变为 ON。
- (5) 当 PTRA 指令已执行一次时，将在第二次和以后碰到 PTRA 指令时将其忽略。
PTRA 指令可通过执行 PTRAR 指令被重新允许。
- (6) 程序跟踪操作的结果可在编程工具上加以监视。

PTRAR

- (1) PTRAR 指令用于复位 PTRA 指令。
- (2) 当执行 PTRAR 时，SM811 到 SM815 变为 OFF。
- (3) 当执行 PTRAR 指令时，PTRA 再次被允许执行。

PTRAEXE

- (1) 当程序跟踪命令变为 ON 时，执行程序跟踪。
- (2) 如果 SM811 在执行程序跟踪期间变为 OFF，将暂停采样。
- (3) 如果程序跟踪命令为 OFF，将不执行任何操作。

备注

- (1) 有关程序跟踪操作的详细信息，请参阅使用的 CPU 模块的用户手册。
- (2) 有关在编程工具上执行程序跟踪的信息，请参阅编程工具的操作手册。

[运行错误]

- (1) 没有和 PTRAEXE (P), PTRAR 指令相关的运行错误。

[程序示例]

- (1) 当 X0 变为 ON 时，以下程序执行 PTRAR 指令，并且当 X1 变为 ON 时，使用 PTRAR 指令复位 PTRAR 指令。

[梯形图模式]

```
graph TD
    subgraph Ladder [ ]
        direction TB
        R0["0 | X0 |-----| [ PTRAR ] |"]
        R2["2 | X1 |-----| [ PTRAR ] |"]
        R4["4 |-----| [ END ] |"]
    end
```

[列表模式]

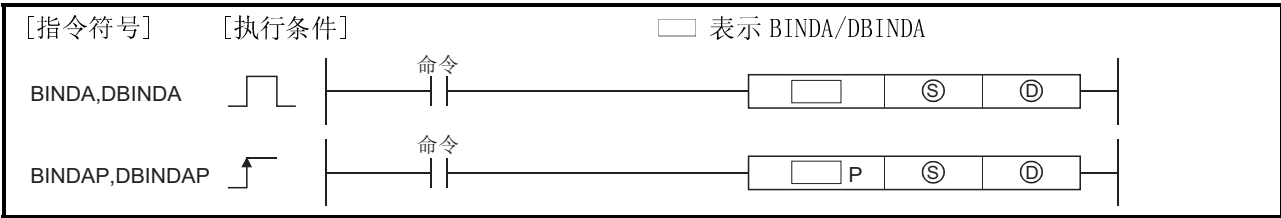
步	指令	软元件
0	LD	X0
1	PTRAR	
2	LD	X1
3	PTRAR	
4	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.11 字符串处理指令

7.11.1 从 BIN 16 位或 32 位到十进制 ASCII 的转换
(BINDA, BINDAP, DBINDA, DBINDAP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J□□□		特殊功能模 块 U□□□□□	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○	○			○				—
⑥	—	○			—				—

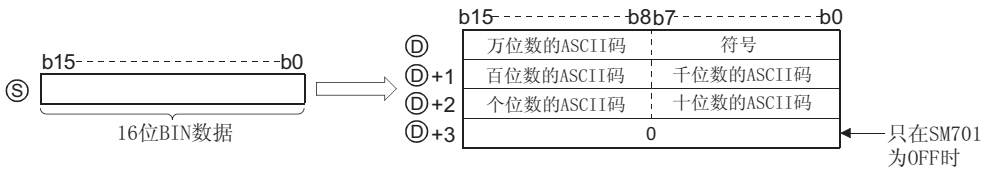


[设定数据]

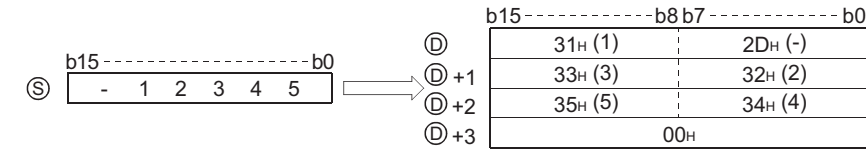
设定数据	含义	数据类型
⑤	要转换为 ASCII 的 BIN 数据	BIN 16/32 位
⑥	存储转换结果的软元件的起始号	字符串

[功能]

- BINDA
- (1) 将每个由⑤指定的以十六进制显示的 BIN 16 位数据的单个号码转换成其各自的 ASCII 码，并从⑥指定号码的软元件开始执行存储。



例如，如果-12345 已经在⑤上被指定，则以下内容将被从⑥向前存储起来：



- (2) 在⑤上指定的 BIN 数据可在从-32768 到 32767 的范围内。

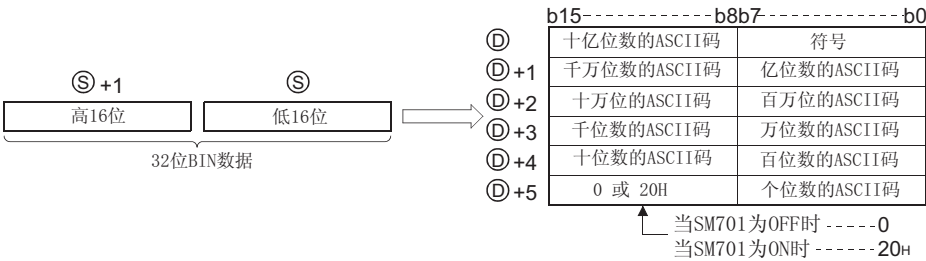
- (3) 存储在⑤上的操作结果如下：
- ① 如果 BIN 数据为正，则将存储符号“20_H”，如果为负，则存储符号“2D_H”。
 - ② 对有效位数前面的零，存储符号“20_H”。(执行零压缩。)

0 0 3 2 5
↑ 有效位数
变为20_H

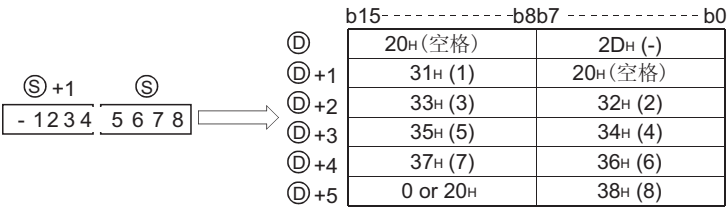
- ③ 由⑤+3 指定的软元件上存储的数据依据 SM701 (字符转换信号的输出号) 的 ON/OFF 状态会有所不同。
- 当 SM701 为 OFF 时..... 存储“0”
- 当 SM701 为 ON 时..... 不变

DBINDA

- (1) 当以十进制符号显示时，在每个位置为每个号码将⑤指定的 BIN 32 位数据转换成 ASCII 码，并将其存储在⑤指定的软元件之后。



例如，如果值-12345678 已经被⑤指定，以下内容将被存储在⑤之后：



- (2) 由⑤指定的 BIN 数据可以在-2147483648 到 2147483647 之间。
- (3) 在⑤上存储的运行结果将以下面的方法执行存储：
- ① 如果 BIN 数据为正，则存储符号“20_H”，如果为负，则存储符号“2D_H”。
 - ② 对有效位数前面的零，存储符号“20_H”。(执行零压缩。)

0 0 1 2 0 3 4 5 6 0
↑ 有效位数
变为20_H

- ③ 在⑤+5 指定的软元件的高 8 位上存储的数据依据 SM701 (输出选择信号的字符数) 的 ON/OFF 状态，而有所不同。
- 当 SM701 为 OFF 时 存储 “0”
- 当 SM701 为 ON 时 存储 “20H”

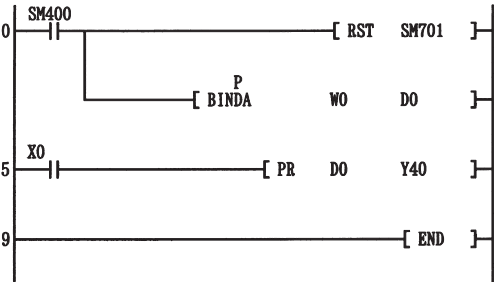
[运行错误]

没有和 BINDA (P) 或 DBINDA (P) 指令相关的运行错误。

[程序示例]

- (1) 以下示例程序使用 PR 指令将 W0 上的 16 位 BIN 数据的十进制值转换乘 ASCII 码，并输出到 Y40 到 Y48。

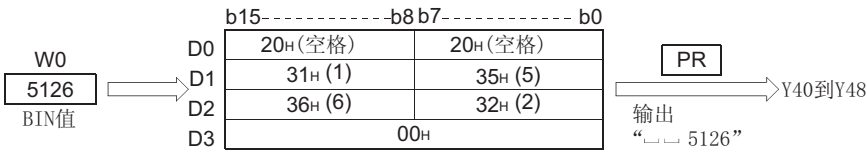
[梯形图模式]



[列表模式]

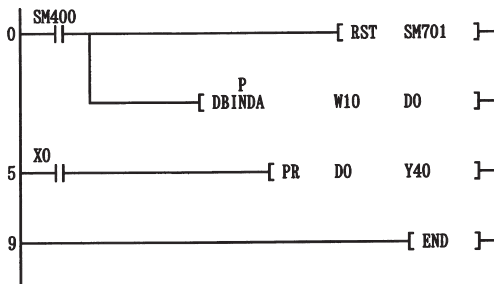
步	指令	软元件
0	LD	SM400
1	RST	SM701
2	BINDAP	W0 D0
5	LD	X0
6	PR	D0 Y40
9	END	

- * 当 X0 变为 ON 时，使用 PR 指令执行 Y40 到 Y48 的 ASCII 输出。
- 由于 SM701 为 OFF，所以 PR 指令将输出 ASCII 码，直到碰到 00H。



- (2) 以下程序使用 PR 指令将 W10 和 W11 上的 32 位 BIN 数据的十进制值转换成 ASCII 码，并输出到 Y40 到 Y48 上。

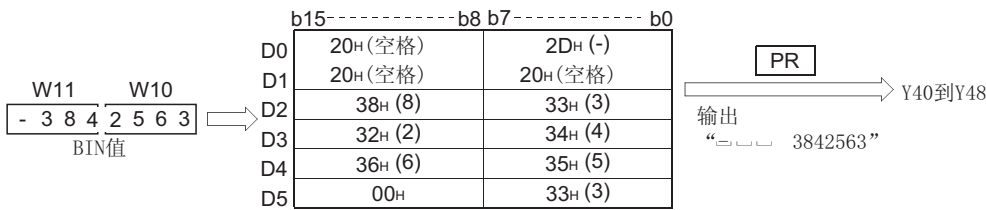
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	RST	SM701
2	DBINDAP	W10 D0
5	LD	X0
6	PR	D0 Y40
9	END	

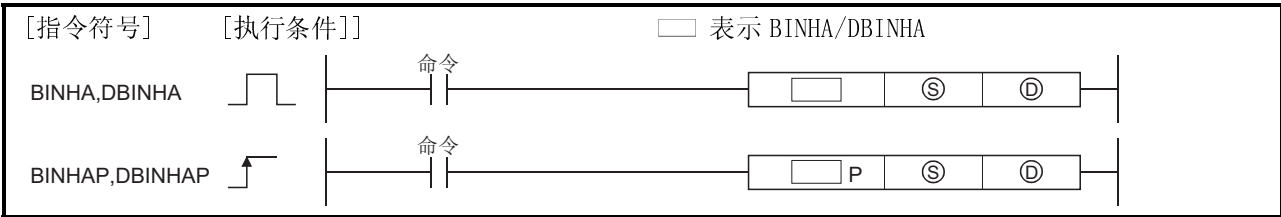
- * 当 X0 变为 ON 时，使用 PR 指令执行 Y40 到 Y48 的 ASCII 输出。
- 由于 SM701 为 OFF，所以 PR 指令将输出 ASCII 码，直到碰到 00H。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7. 11. 2 从 BIN 16 位或 32 位数据到十六进制 ASCII 的转换
(BINHA, BINHAP, DBINHA, DBINHAP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	○	○			○		—
⑥	—	○			—		—



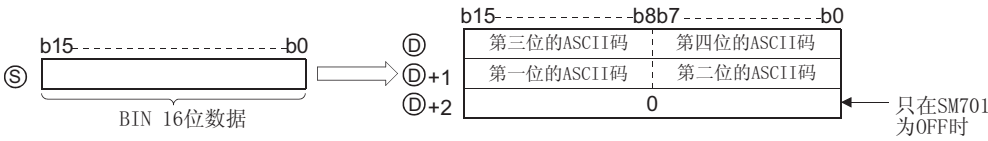
[设定数据]

设定数据	含义	数据类型
⑤	要转换为 ASCII 的 BIN 数据	BIN 16/32 位
⑥	存储转换结果的软元件的起始号	字符串

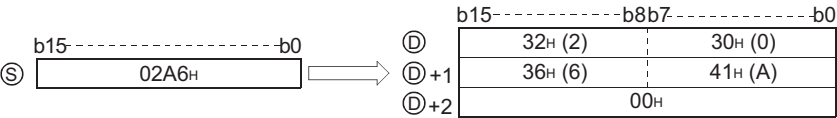
[功能]

BINHA

(1) 将由⑤指定的以十六进制显示的 BIN 16 位数据的每个号码转换成其各自的 ASCII 码，并从⑥指定号码的软元件开始存储。



例如，如果 02A6H 已经被⑤指定，则将按照如下所示执行存储：

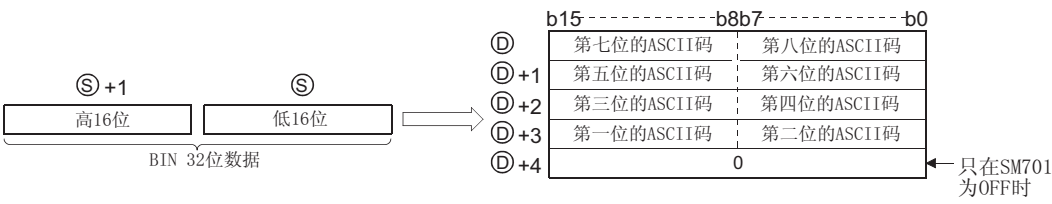


- (2) 由⑤指定的 BIN 数据可在 0H 到 FFFFH 的范围内。
- (3) 在⑥上存储的运行结果被作为 4 位十六进制值处理。
因此，位于该值左边的作为有效位的零被作为“0”处理。（未执行零压缩。）

- (4) 在D+2 指定的软元件上存储的数据依据 SM701 (输出选择信号的字符数) 的 ON/OFF 状态将会有所不同。
- 当 SM701 为 OFF 时 存储 “0”
- 当 SM701 为 ON 时 不变

DBINHA

- (1) 当以十进制符号显示时，在每个位置为每个号将由D指定的 BIN 32 位数据转换成 ASCII 码，并将其存储在由D指定的软元件之后。



例如，如果值 03AC625E_H 已经被D指定，则将其按照下面的方式存储在D之后：



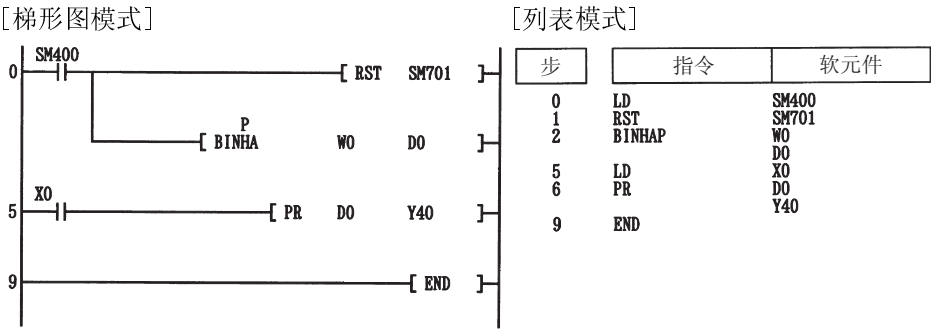
- (2) 由D指定的 BIN 数据可在 0_H 到 FFFFFFFF_H 的范围内。
- (3) 在D上存储的运行结果被作为 8 位十六进制值处理。
- 因此，位于该值左边的作为有效位的零被作为 “0” 处理。(未执行零压缩。)
- (4) 在D+2 指定的软元件上存储的数据依据 SM701 (输出选择信号的字符数) 的 ON/OFF 状态将会有所不同。
- 当 SM701 为 OFF 时 存储 “0”
- 当 SM701 为 ON 时 不变

[运行错误]

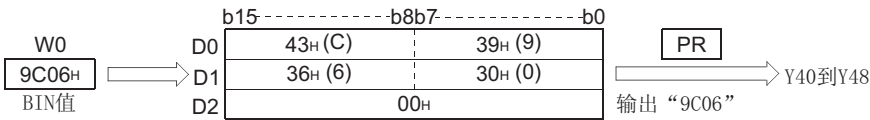
没有和 BINHA (P) 或 DBINHA (P) 指令相关的运行错误。

[程序示例]

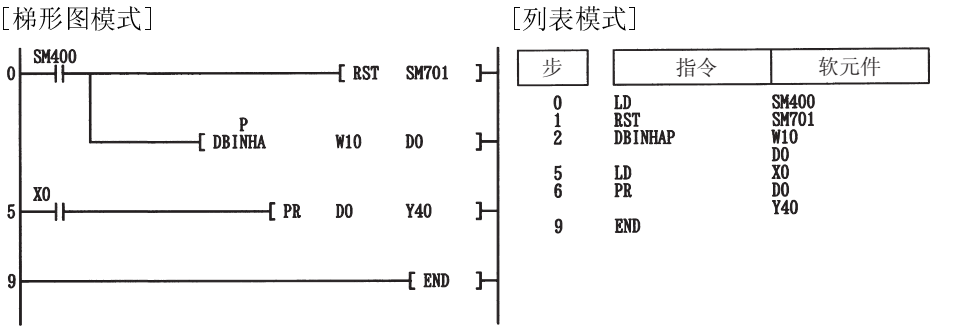
(1) 以下程序使用 PR 指令将 W0 上的 16 位 BIN 数据的十六进制值转换成 ASCII 码形式，并输出到 Y40 到 Y48 上。



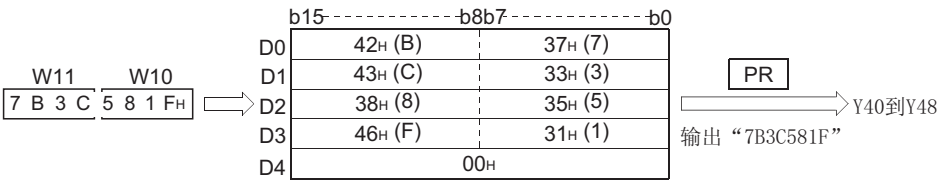
※ 当 X0 变为 ON 时，通过 PR 指令将 ASCII 码输出到 Y40 到 Y48 上。
由于 SM701 为 OFF，PR 指令将输出 ASCII 码，直到碰到 00H。



(2) 以下程序使用 PR 指令将 W10 和 W11 上的 32 位 BIN 数据的十六进制值转换成 ASCII 码形式，并输出到 Y40 到 Y48 上。



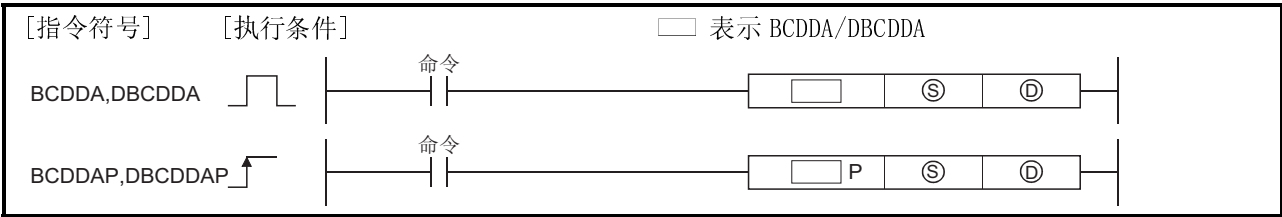
※ 当 X0 变为 ON 时，通过 PR 指令将 ASCII 码输出到 Y40 到 Y48 上。
由于 SM701 为 OFF，PR 指令将输出 ASCII 码，直到碰到 00H。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.11.3 从 BCD 4 位和 8 位数据到十进制 ASCII 数据的转换
(BCDDA, BCDDAP, DBCDDA, DBCDDAP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][]G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤	○	○			○				—
⑩	—	○			—				—



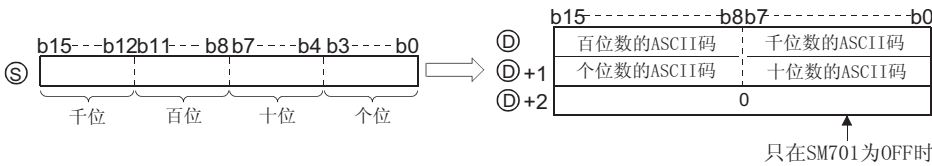
[设定数据]

设定数据	含义	数据类型
⑤	将被转换成 ASCII 码的 BCD 数据	BCD 4 位/8 位
⑩	存储转换结果的软元件的起始号	字符串

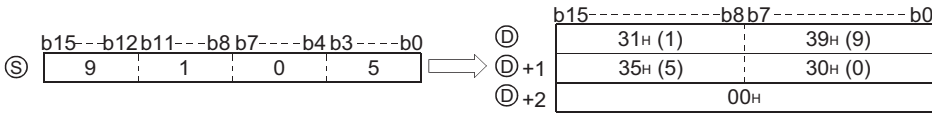
[功能]

BCDDA

(1) 将⑤指定的 BCD 4 位数据中的各个号码转换成它们各自的 ASCII 码，并从⑩指定的软元件开始存储。



例如，如果⑤指定了值 9105，则将以下面的方式将运行结果存储在⑩上：



(2) ⑤指定的 BCD 数据可以在 0 到 9999 的范围内。

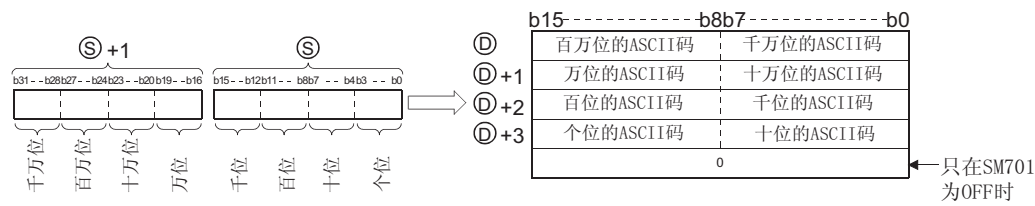
(3) 计算结果存储在软元件⑩上。“有效位数”左侧的所有零都被压缩。



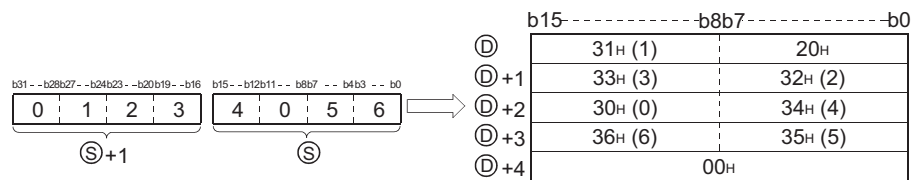
- (4) 要在⑩+2 指定的软元件上存储的数据依据 SM701 (输出选择信号的字符数) 的 ON/OFF 状态将会有所不同。
- 当 SM701 为 OFF 时 存储 “0”
- 当 SM701 为 ON 时 不变

DBCDDA

- (1) 将⑤指定的 BCD 8 位数据中的各个号码转换成它们各自的 ASCII 码，并存储在⑩指定的软元件之后。



例如，如果值 01234056 已经被⑤指定，则将按照以下方式将运行结果存储在⑩之后：



- (2) ⑤指定的 BCD 数据可以在 0 到 99999999 的范围内。
- (3) 计算结果存储在软元件⑩上，“有效位数”左侧的所有零都被压缩。



- (4) 要在⑩+4 指定的软元件上存储的数据依据 SM701 (输出选择信号的字符数) 的 ON/OFF 状态将会有所不同。
- 当 SM701 为 OFF 时 存储 “0”
- 当 SM701 为 ON 时 不变

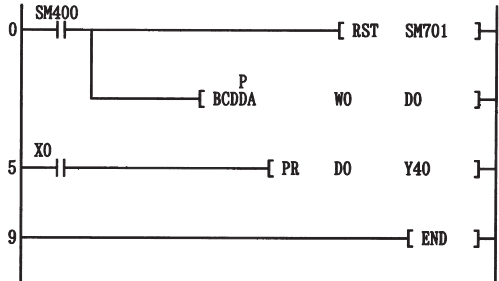
[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 上。
- ⑤上的数据在 DBCDDA 指令操作期间超出了 0 到 9999 的范围。 (错误代码: 4100)
 - ⑤上的数据在 DBCDDA 指令操作期间超出了 0 到 99999999 的范围。 (错误代码: 4100)

[程序示例]

(1) 以下程序使用 PR 指令将 BCD 4 位数据 (W0 上的值) 转换成十进制数，并以 ASCII 格式将其输出到 Y40 到 Y48 上。

[梯形图模式]

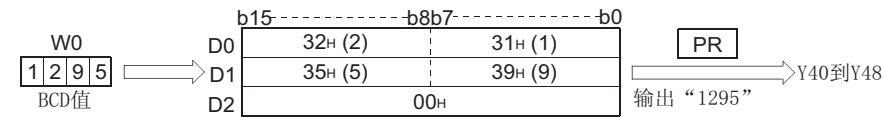


[列表模式]

步	指令	软元件
0	LD	SM400
1	RST	SM701
2	BCDDAP	W0 D0
5	LD	X0
6	PR	D0 Y40
9	END	

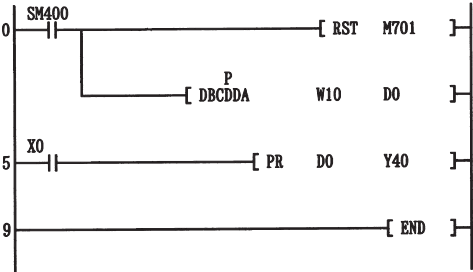
[操作]

当 X0 变为 ON 时，使用 PR 指令执行 Y40 到 Y48 的 ASCII 码输出。
由于 SM701 为 OFF，PR 指令将输出 ASCII 码，直到碰到 00H。



(2) 以下程序使用 PR 指令将 BCD 8 位数据 (W0 和 W11 上的值) 转换成十进制数，并以 ASCII 格式将其输出到 Y40 到 Y48 上。

[梯形图模式]

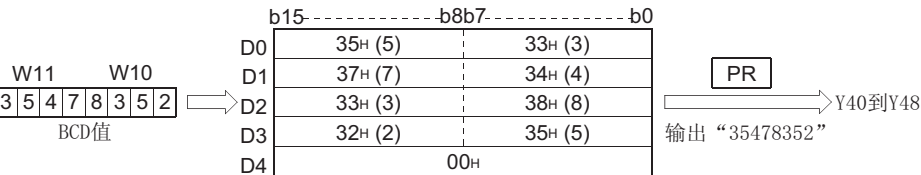


[列表模式]

步	指令	软元件
0	LD	SM400
1	RST	M701
2	DBCDDAP	W10 D0
5	LD	X0
6	PR	D0 Y40
9	END	

[操作]

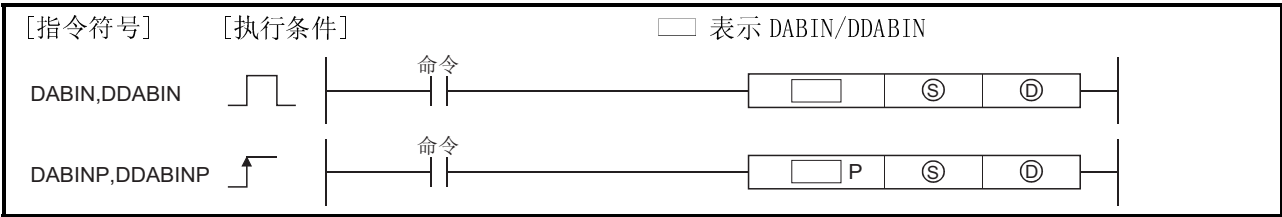
当 X0 变为 ON 时，使用 PR 指令执行 Y40 到 Y48 的 ASCII 码输出。
由于 SM701 为 OFF，PR 指令将输出 ASCII 码，直到碰到 00H。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.11.4 从十进制 ASCII 码到 BIN 16 位和 32 位数据的转换
(DABIN, DABINP, DDABIN, DDABINP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [Z] \ [Z]		变址寄存器 Zn	常量 \$
	位	字		位	字		
⑤	—	○		—			○
④	○	○		○			—



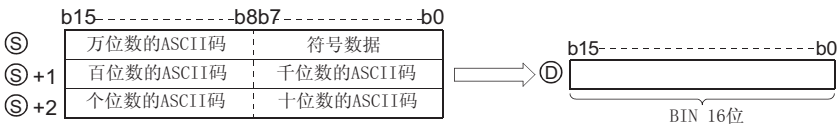
[设定数据]

设定数据	含义	数据类型
⑤	存储将被转换成 BIN 值的 ASCII 数据的软元件的起始号	字符串
④	存储转换结果的软元件的起始号	BIN 16/32 位

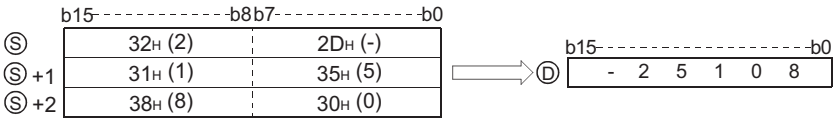
[功能]

DABIN

(1) 将存储在由⑤指定号码的软元件之后的十进制 ASCII 数据转换成 BIN 16 位数据，并将其存储在由④指定号码的软元件中。



例如，如果⑤之后的 ASCII 码被指定为值-25018H，则将按照以下方式将运行结果存储在 ④上：

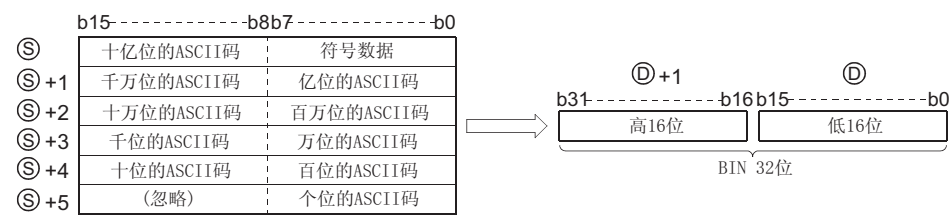


(2) ⑤到⑤+2 指定的 ASCII 数据可以在-32768 到 32767 的范围内。

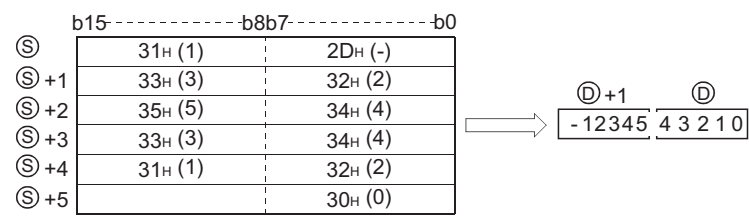
- (3) 如果 BIN 数据为正，则存储符号“20_H”，如果为负，则存储符号“2D_H”。
- (4) 可以在“30_H”到“39_H”的范围内给每个位置设置 ASCII 码。
- (5) 如果给单个位置设置的 ASCII 码是“20_H”或“00_H”，则将作为“30_H”处理。

DDABIN

- (1) 将存储在⑤指定号码的软元件之后的十进制 ASCII 数据转换成 BIN 32 位数据，并将其存储在⑤+1指定号码的软元件中。



例如，如果⑤后的 ASCII 码被指定为值-1234543210_H，则将按照以下方式将运行结果存储在⑤+1 和⑤上：



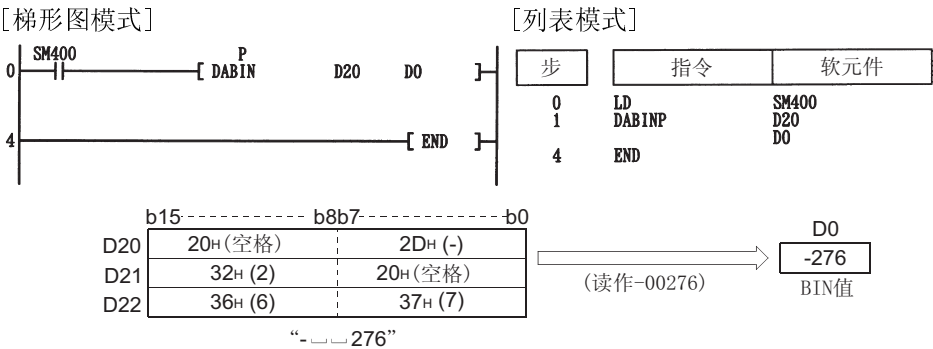
- (2) 由⑤到⑤+5 指定的 ASCII 数据可以在-2147483648 到 2147483647 的范围内。
此外，存储在⑤+5 的高字节上的数据将被忽略。
- (3) 如果转换数据的值为正，则“符号”数据是“20_H”，如果为负，则是“2D_H”。
- (4) 可以在“30_H”到“39_H”的范围内给每个位置设置 ASCII 码。
- (5) 如果给单个位置设置的 ASCII 码是“20_H”或“00_H”，则将作为“30_H”处理。

[运行错误]

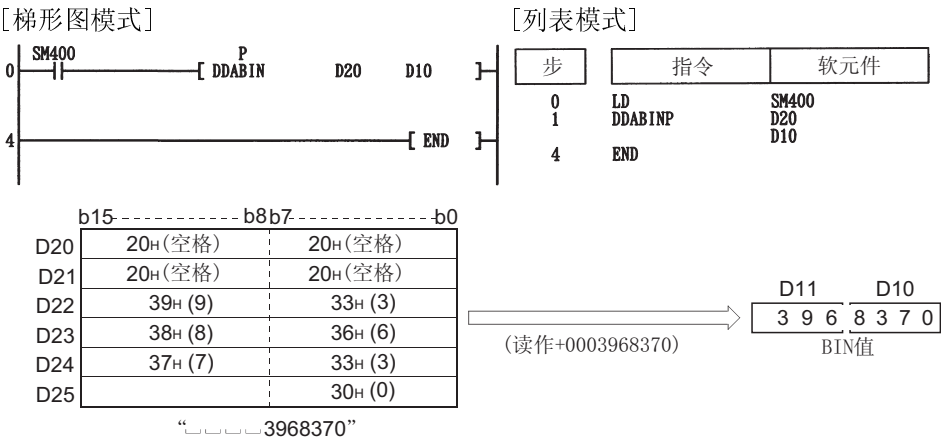
- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 上。
- ⑤指定的符号数据不是“20H”或“2Dh”。 (错误代码：4100)
 - ⑤到⑤+5 为单个号码指定的 ASCII 码是“30H”到“39H”，“20H”或“00H”以外的数。 (错误代码：4100)
 - ⑤到⑤+5 指定的 ASCII 码超出了以下所示的范围： (错误代码：4100)
 - 当使用 DABIN 指令时 -32768 到 32767
 - 当使用 DDABIN 指令时 -2147483648 到 2147483647

[程序示例]

- (1) 以下程序将设置在 D20 到 D22 上的十进制, 5 位 ASCII 数据和符号转换成 BIN 值，并将结果存储在 D0 上。



- (2) 以下程序将设置在 D20 到 D25 上的十进制, 10 位 ASCII 数据和符号转换成 BIN 值，并将结果存储在 D10 和 D11 上。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.11.5 从十六进制 ASCII 到 BIN 16 位和 32 位数据的转换
(HABIN, HABINP, DHABIN, DHABINP)

设定数据	可用软元件					
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [J K L]		其它
	位	字		位	字	
⑤	—	○		—		—
⑥	○	○		○		—



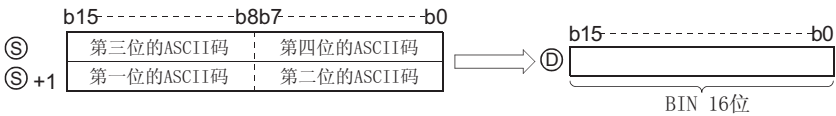
[设定数据]

设定数据	含义	数据类型
⑤	存储将被转换成 BIN 值的 ASCII 数据的软元件的起始号	字符串
⑥	存储转换结果的软元件的起始号	BIN 16/32 位

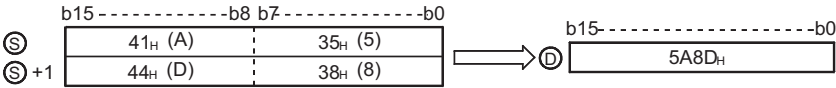
[功能]

HABIN

(1) 将存储在由⑤指定号码的软元件之后的 ASCII 数据转换成 BIN 16 位数据，并将其存储在由⑥指定号码的软元件中。



例如，如果⑤之后的 ASCII 码被指定为值 5A8D_H，则将按照以下方式将运行结果存储在⑥上：

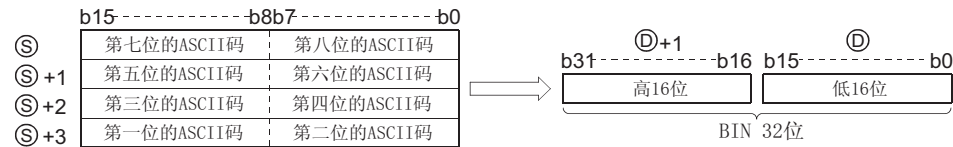


(2) ⑤到⑤+1 指定的 ASCII 数据可以在 0000_H 到 FFFF_H 的范围内。

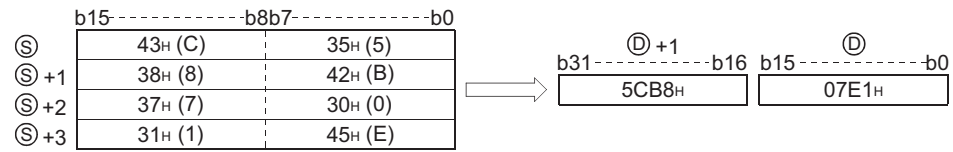
(3) ASCII 码可以在 “30_H” 到 “39_H” 和 “41_H” 到 “46_H” 的范围内。

DHABIN

(1) 将存储在由⑤指定号码的软元件之后的十六进制 ASCII 数据转换成 BIN 32 位数据，并将其存储在由⑥指定号码的软元件中。



例如，如果⑤之后的 ASCII 码被指定为值 5CB807E1_H，则将按照以下方式将运行结果存储在⑥+1 和⑥上：



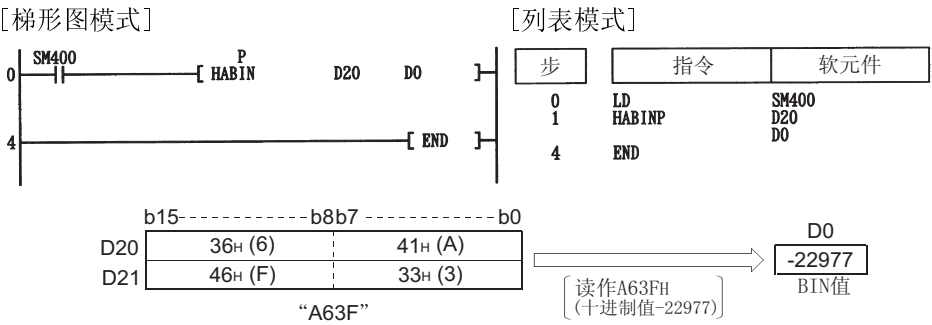
- (2) ⑤到⑤+3 指定的 ASCII 数据可以在 00000000_H 到 FFFFFFFF_H 的范围内。
- (3) ASCII 码可以在“30_H”到“39_H”和“41_H”到“46_H”的范围内。

[运行错误]

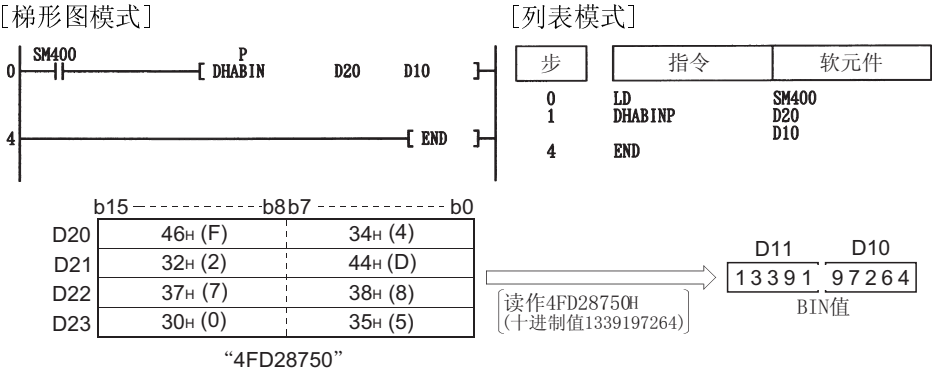
- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 上。
- ⑤和⑤+3 为各个号码指定的 ASCII 码超出了“30_H”到“39_H”和“41_H”到“46_H”的范围。
(错误代码：4100)

[程序示例]

(1) 以下程序将设置在 D20 和 D21 上的十六进制, 4 位 ASCII 数据转换成 BIN 值，并将结果存储在 D0 上。



(2) 以下程序将设置在 D20 到 D23 上的十六进制, 8 位 ASCII 数据转换成 BIN 值，并将结果存储在 D10 和 D11 上。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.11.6 从十进制 ASCII 到 BCD 4 位或 8 位数据的转换
(DABCD, DABCDP, DDABCD, DDABCDP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J K L		特殊功能模 块 U V G	变址寄存器 Zn	常量 \$	其它
	位	字		位	字				
⑤	—	○		—				○	—
⑥	○	○		○				—	—



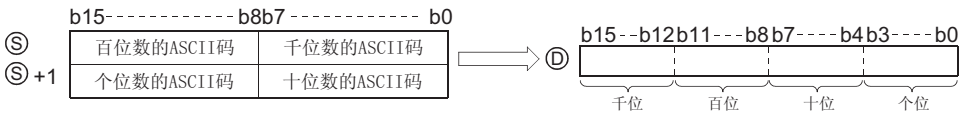
[设定数据]

设定数据	含义	数据类型
⑤	存储将被转换成 BCD 值的 ASCII 数据的软元件的起始号	字符串
⑥	存储转换结果的软元件的起始号	BCD 4 位/8 位

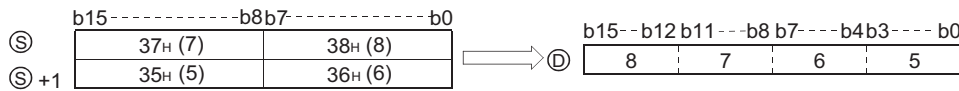
[功能]

DABCD

- (1) 将存储在⑤指定号码的软元件之后的十六进制 ASCII 数据转换成 4 位 BCD 数据，并将其存储在⑥指定号码的软元件上。



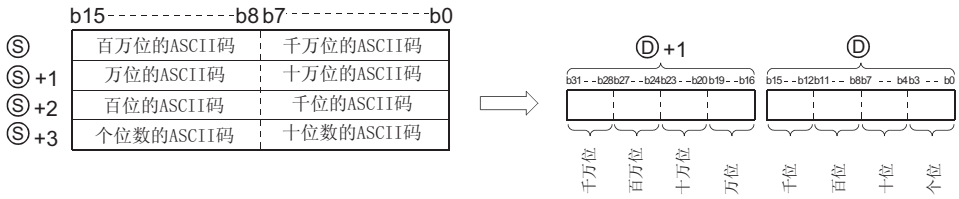
例如，如果⑤之后的 ASCII 码被指定为值 8765_H，则将按照以下方式将运行结果存储在⑥上：



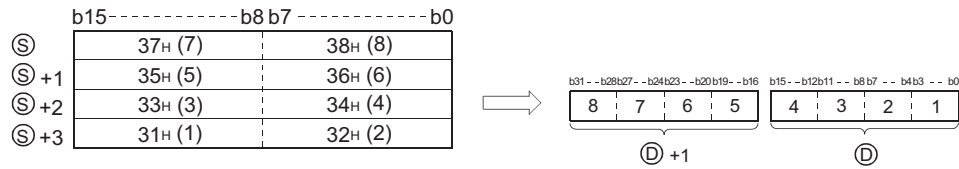
- (2) ⑤到⑤+1 指定的 ASCII 数据可以在 0 到 9999 的范围内。
- (3) 为每个位设置的 ASCII 码可以在“30_H”到“39_H”的范围内。
- (4) 如果单个位的 ASCII 码为“20_H”或“00_H”，则将当作“30_H”处理。

DDABCD

(1) 将存储在由⑤指定号码的软元件之后的十六进制 ASCII 数据转换成 8 位 BCD 数据，并将其存储在由⑥指定号码的软元件上。



例如，如果⑤之后的 ASCII 码被指定为值 87654321_H，则将按照以下方式将运行结果存储在⑥+1 和⑥上：



- (2) ⑤到⑤+3 指定的 ASCII 数据可以在 0 到 99999999 的范围内。
- (3) 为每个位设置的 ASCII 码可以在“30_H”到“39_H”的范围内。
- (4) 如果单个位的 ASCII 码是“20_H”到“00_H”，则将当作“30_H”处理。

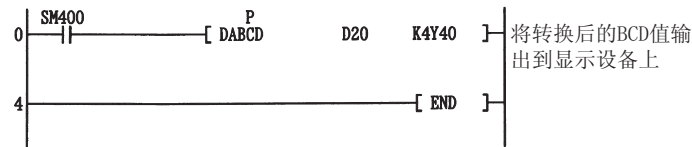
[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 上。
- 在⑤上的数据内，有字符超出了 0 到 9 的范围。(错误代码：4100)

[程序示例]

(1) 以下程序将 D20 到 D22 上设置的十六进制 ASCII 数据转换成 BCD 4 位数据，并将结果输出到 Y40 到 Y4F 上。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	DABCDP	D20
		K4Y40
4	END	

b15-----b8b7-----b0	
D20 34H (4) 20H (space)	
D21 34H (4) 39H (9)	

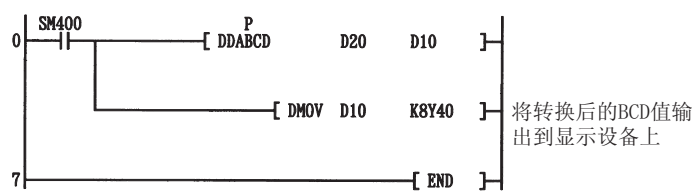
“ 494 ”

读作0494

Y4F--Y40 BCD值

(2) 以下程序将设置在 D20 到 D22 上的十六进制 ASCII 数据转换成 8 位 BCD 数据，将结果存储在 D10 和 D11 上，并将其输出到 Y40 到 Y5F 上。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	DDABCDP	D20
		D10
4	DMOV	D10
		K8Y40
7	END	

b15-----b8b7-----b0	
D20 34H (4) 20H (空格)	
D21 37H (7) 39H (9)	
D22 39H (9) 32H (2)	
D23 39H (9) 34H (4)	

“ 4972949 ”

读作04972949

D11 D10 BCD值

DMOV

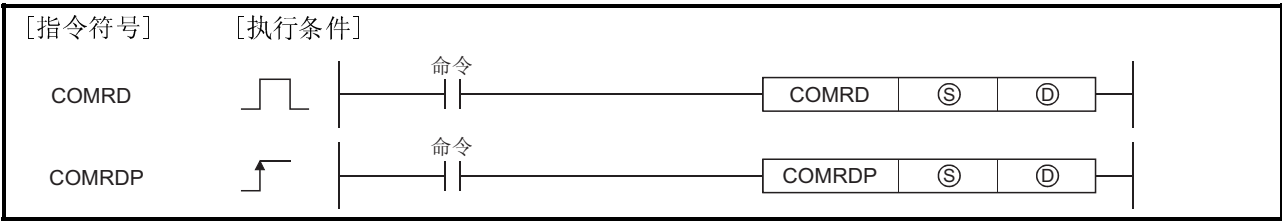
Y5F--Y50Y4F--Y40 BCD值

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.11.7 读取软元件注释数据 (COMRD, COMRDP)

(1) 当使用高性能型 QCPU/过程控制 CPU 时

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\K[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				BL\S, BL\TR, BL, P, I, J, U
⑤	○	○		○			—		○
⑥	—	○		—			—		—

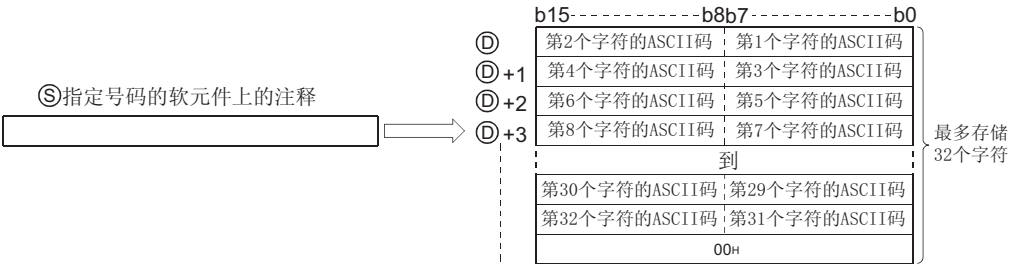


[设定数据]

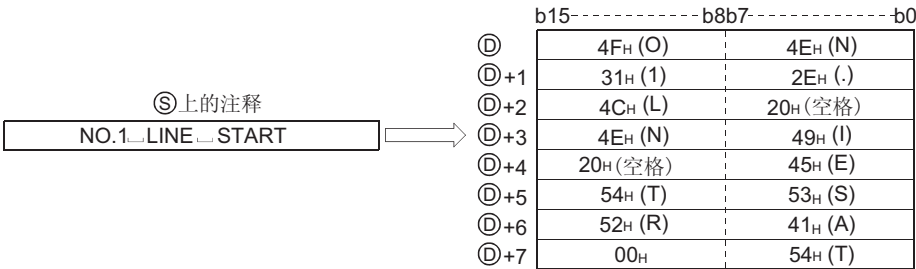
设定数据	含义	数据类型
⑤	登记将被读取的注释的软元件的起始号	软元件名称
⑥	存储已读取的注释数据的软元件的起始号	字符串

[功能]

(1) 读取由⑤指定的软元件上的注释，并以将其以 ASCII 码存储在⑥指定号码的软元件之后。



例如，如果指定软元件⑤的注释是“NO.1 LINE START”，则运行结果将被存储在⑥之后，如下：



- (2) 能被⑤指定号码的软元件仅限于在注释范围内设定的那些号码。
- (3) 如果没有给⑤指定的软元件登记注释，则该注释的所有字符将被处理为“20_H”（空格）。
- (4) 存储最后的⑤字符的软元件号加 1 将依据 SM701 (输出选择信号的字符数) 的 ON/OFF 状态而有所不同。
- 当 SM701 为 OFF 时：存储“0”
- 当 SM701 为 ON 时：不变
- (5) 当读取注释时，该指令完成后 SM720 变为 ON 一个扫描周期，SM721 在该指令执行期间变为 ON。
- 当 SM721 为 ON 时，无法执行 COMRD (P) 指令，如果尝试执行，将不作处理。

要点
(1) 对于 COMRD (P) 指令使用的软元件注释，使用存储在存储卡中的注释文件。 不能使用程序存储器中存储的注释文件。
(2) 在参数模式下的“PLC 文件设置”中设置 COMRD (P) 指令使用的注释。 如果还未在 PLC 文件设置中设置使用的注释文件，则 COMRD (P) 指令无法输出软元件注释。
(3) 不要在中断程序中使用 COMRD (P) 指令。 否则，将导致产生故障。

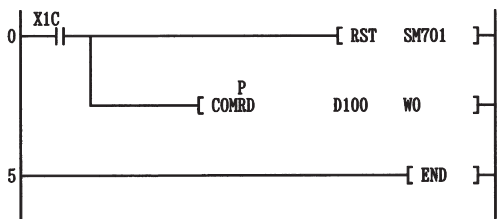
[运行错误]

- (1) 在以下情形中，返回一个错误，并且错误标志变为 ON。
- ⑤指定号码的软元件不在注释范围设置之内。(错误代码：4100)
 - 当在 RUN 期间写入注释时执行 COMRD (P) 指令。(错误代码：4100)
 - 指定的文件不存在。(错误代码：2410)

[程序示例]

(1) 当 X1C 变为 ON 时，以下程序将 D100 上设置的注释以 ASCII 存储在 W0 中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X1C
1	RST	SM701
2	COMRDP	D100 W0
5	END	

W0	4C _H (L)	41 _H (A)
W1	4E _H (N)	49 _H (I)
W2	20 _H (空格)	45 _H (E)
W3	41 _H (A)	54 _H (T)
W4	57 _H (G)	52 _H (R)
W5	54 _H (T)	55 _H (E)
W6	20 _H (空格)	20 _H (空格)
W7	00 _H	

D100上的注释。
ALINE □ TARGET □

备注

- (1) 高性能型 QCPU 和过程控制 CPU 在几次扫描之后完成处理。QnACPU 则立即完成处理。
- (2) 如果在该指令结束前(当 SM720 为 ON 时)将 COMRD(P)/PRC 指令的启动信号(命令)变为 ON, 则不执行 COMRD(P)/PRC 指令。当 SM721 为 OFF 时, 执行 COMRD(P)/PRC 指令。
- (3) 不能同时访问两个或两个以上的文件注释。
- (4) 以下指令由于共同使用 SM721, 所以不能同时执行。

指令名	执行期间 ON	结束后 ON 一个扫描周期	异常结束后 ON
S. FREAD S. FWRITE	SM721	由指令指定	(指令指定的软元件)+1
PRC COMRD		SM720	无

(2) 当使用 QnACPU 时

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				BL\S, BL\TR, BL, P, I, J, U
⑤	○	○		○			—		○
⑥	—	○		—			—		—

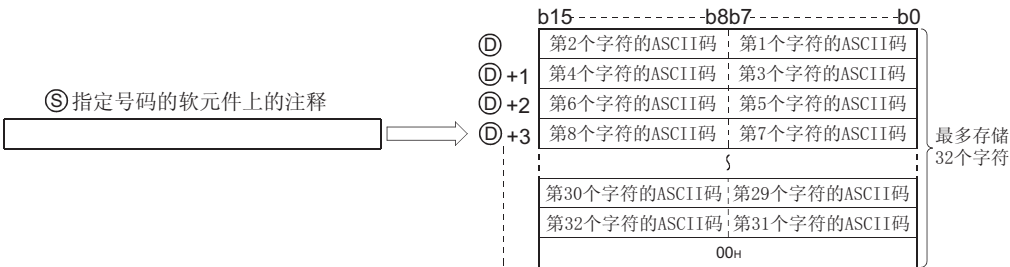


[设定数据]

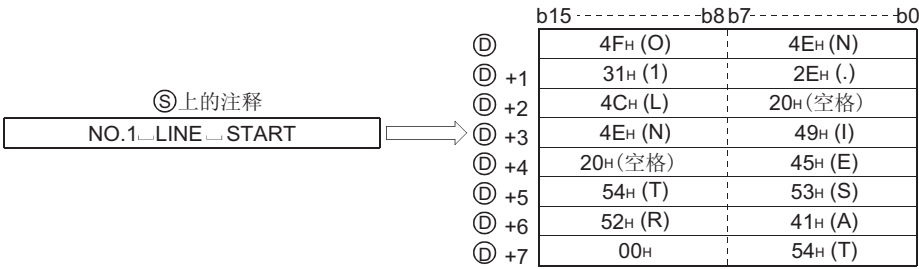
设定数据	含义	数据类型
Ⓢ	登记即将读取的注释的软元件的起始号	软元件名称
Ⓓ	存储已读取的注释数据的软元件的起始号	字符串

[功能]

(1) 读取由Ⓢ指定的软元件上的注释，并将其以 ASCII 码存储在由Ⓓ指定号码的软元件之后。



例如，如果由Ⓢ指定的软元件上的注释是“NO.1 LINE START”，则运行结果将存储在 Ⓓ后，如下：



- (2) 能被Ⓢ指定号码的软元件仅限于注释范围内设定的那些号。
- (3) 如果没有给Ⓢ指定的软元件登记注释，则该注释的所有字符将被处理为“20H”（空格）。

- (4) 存储最后的㊸字符的软元件号加 1，将依据 SM701 (输出选择信号的字符数) 的 ON/OFF 状态而有所不同。
- 当 SM701 为 OFF 时：存储 “0”
- 当 SM701 为 ON 时：不变

要点
(1) COMRD 指令中使用的软元件的注释，使用存储卡中存储的注释文件。 不能使用内置存储器中存储的注释文件。
(2) 使用参数模式中的“PLC 文件设置”来设置 COMRD 指令使用的注释文件。 如果 PLC 文件设置中使用的注释文件还未指定，则软元件注释无法通过 COMRD 指令输出。

[运行错误]

- (1) 在以下情形中，返回一个错误，并且错误标志变为 ON。
- 由㊸指定号码的软元件不在注释范围设置之内。
- (错误代码：4100)

[程序示例]

- (1) 当 X1C 变为 ON 时，以下程序将 D100 上设置的注释以 ASCII 存储在 W0 中。

[梯形图模式]

[列表模式]

步	指令	软元件
0	LD	X1C
1	RST	SM701
2	COMRDP	D100 W0
5	END	

D100上的注释。

ALINE└─TARGET└─

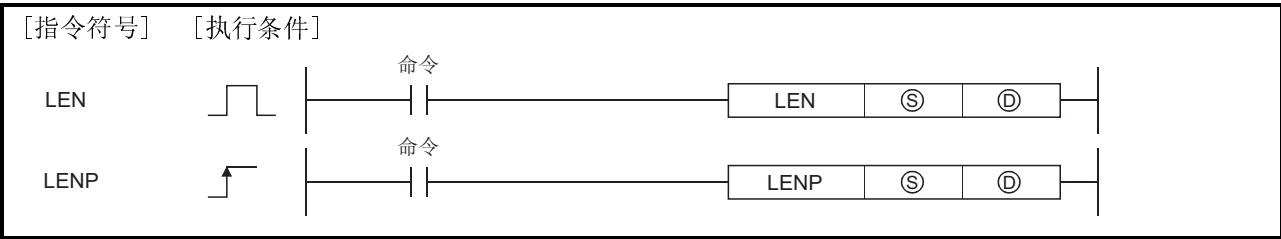
→

	b15-----b8	b7-----b0
W0	4CH (L)	41H (A)
W1	4EH (N)	49H (I)
W2	20H (空格)	45H (E)
W3	41H (A)	54H (T)
W4	57H (G)	52H (R)
W5	54H (T)	55H (E)
W6	20H (空格)	20H (空格)
W7	00H	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.11.8 字符串长度检测 (LEN, LENP)

数据 设定	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○		—			○
⑥	○	○		○			—

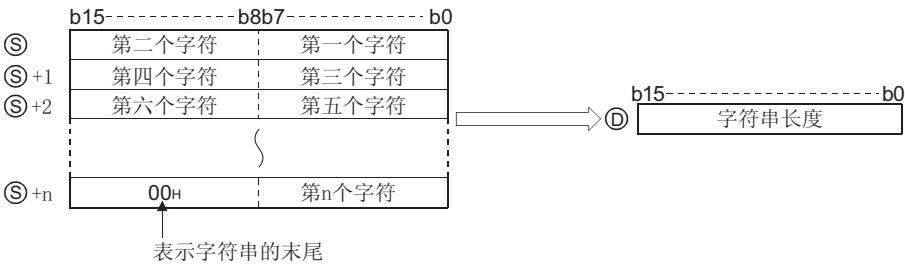


[设定数据]

设定数据	含义	数据类型
⑤	存储字符串的软元件的起始号	字符串
⑥	用来存储已检测字符串长度的软元件数	BIN 16 位

[功能]

- (1) 检测由⑤指定的字符串的长度，并将其存储在由⑥指定号码的软元件之后。
将从由⑤指定号码的软元件开始到存储“00H”的软元件之间的数据作为一个字符串处理。



例如，如果正从⑤向前存储值“ABCDEFGH”，则值9将被存储在⑥上。



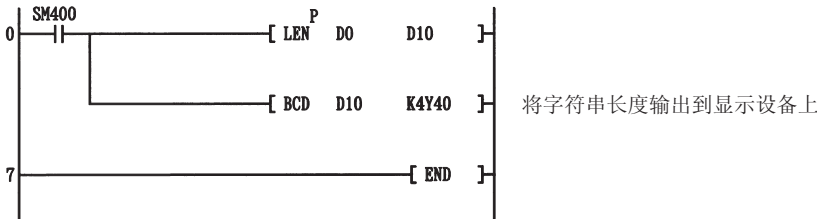
[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 上。
- 在由⑤指定号码的软元件之后的相关软元件范围之内没有设定“00H”。（错误代码：4100）

[程序示例]

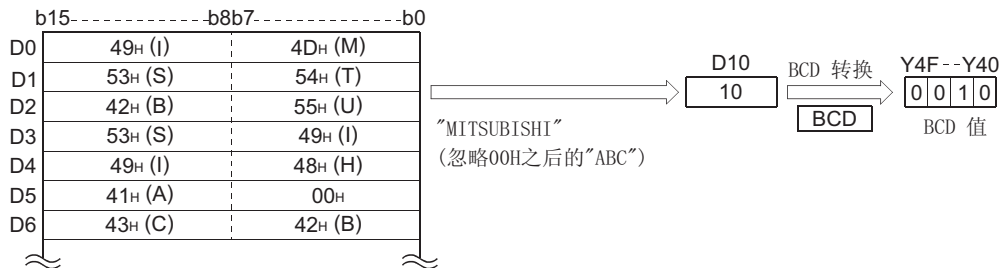
- (1) 以下程序将 D0 上的字符串的长度以 BCD 4 位值输出到 Y40 到 Y4F 上。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	LENP	D0 D10
4	BCD	D10 K4Y40
7	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

*：序列号前五位是04122或更后的。

7.11.9 从 BIN 16 位或 32 位到字符串的转换 (STR, STRP, DSTR, DSTRP)

数据 设定	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
①	○	○			○			—	—
②	○	○			○			○	—
③	—	○			—			—	—

[指令符号]	[执行条件]	
STR, DSTR		命令
STRP, DSTRP		命令

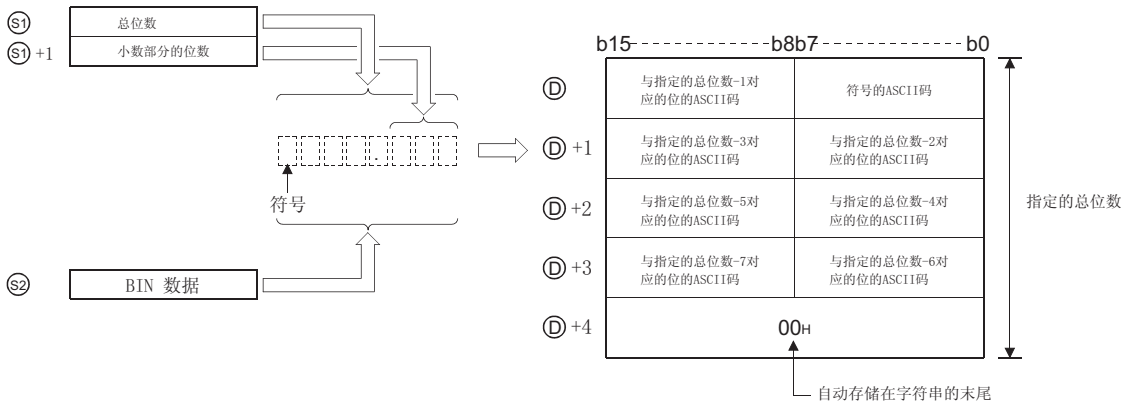
[设定数据]

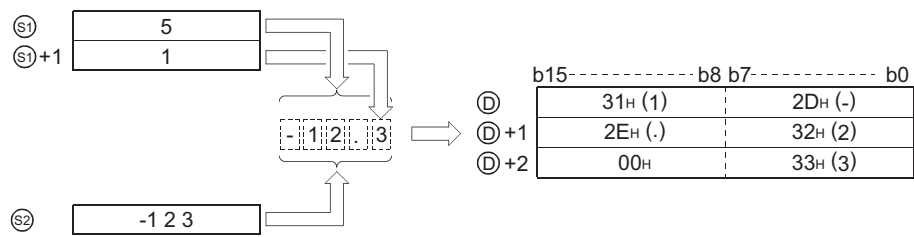
设定数据	含义	数据类型
①	存储将要转换的数值的位数的软元件的起始号	BIN 16 位
②	将被转换的 BIN 数据	BIN 16/32 位
③	存储已转换字符串的软元件的起始号	字符串

[功能]

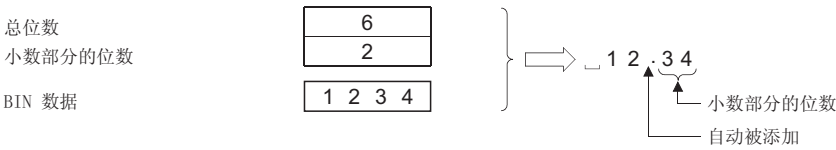
STR

(1) 在由①指定的位置上，给由②指定的 BIN 16 位数据添加一个小数点，并将该数据转换成字符串数据，从由③指定号码的软元件开始将其存储起来。



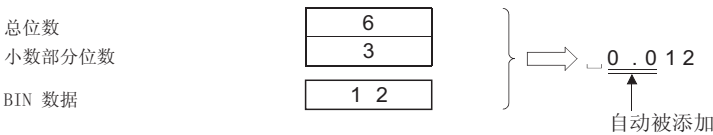


- (2) ①能够指定的总位数是 2 到 8。
- (3) 能够被①+1 指定的作为小数点后的一部分的位数是 0 到 5。
但是，小数点后面的位数必须小于或等于总位数减 3。
- (4) 在-32768 和 32767 之间内的 BIN 数据可在②上指定。
- (5) 转换之后，字符串数据按照下面所示被存储在号码在①之后的软元件上：
- ① 如果 BIN 数据为正，则存储符号“20H”（空格）；如果为负，则存储符号“2DH”（负号-）。
 - ② 如果小数点之后的位数的设置是 0 以外的任何数字，则“2EH”（.）将被自动存储在指定位数的第一位之前的位置。

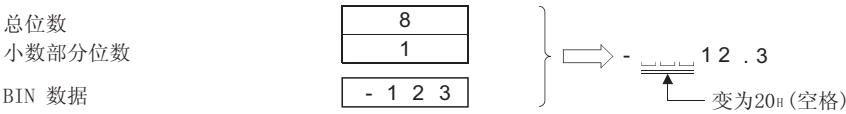


如果小数部分的位数是 0，则该值不存储“2EH”（.）。

- ③ 如果小数点之后的总位数比 BIN 数据的位数大，将自动增加一个零，并通过向右移位来转换该数据，从而变为“0. [] [] [] []”。



④ 如果除了符号和小数点之外的总位数大于 BIN 数据的位数，则“20H”（空格）将被存储在符号和数值之间。

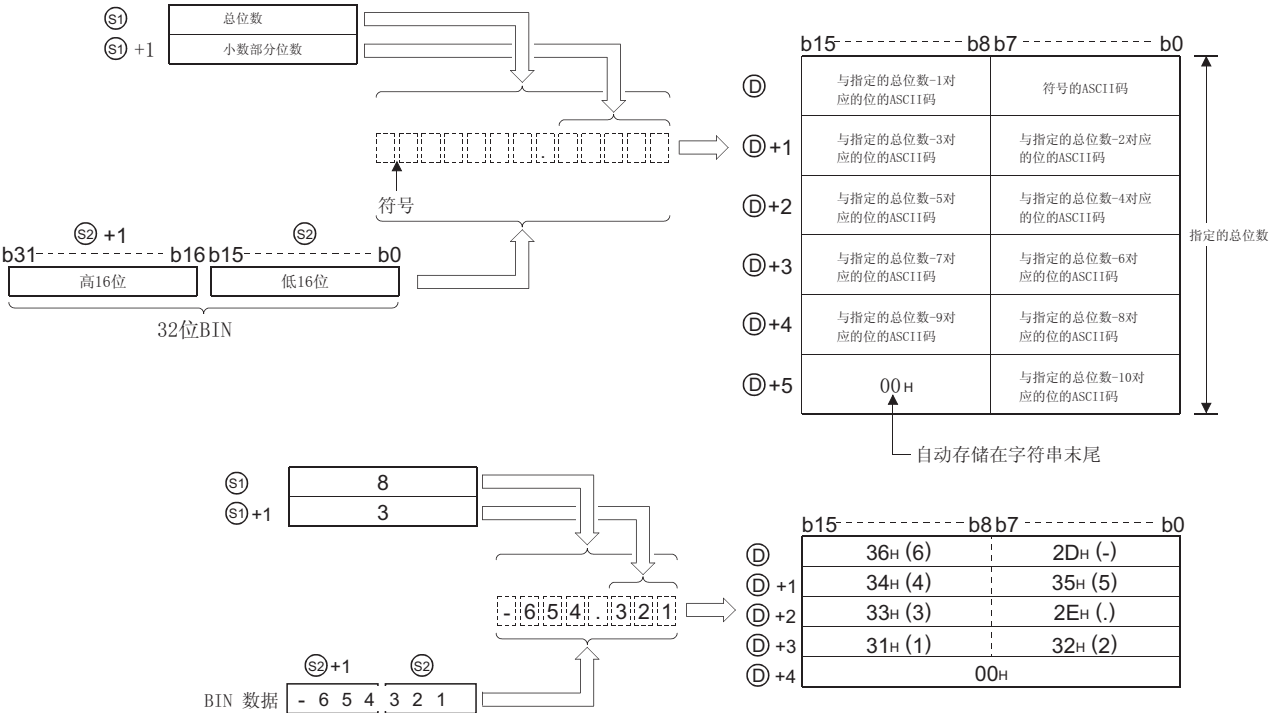


如果 BIN 的位数较大，将返回一个错误。

⑤ 值“00H”被自动存储在已转换字符串的末端。

DSTR

(1) 在由⑤1指定的位置上，给由⑤2指定的 BIN 32 位数据添加一个小数点，将该数据转换成字符串数据，并将其存储在由⑤指定号码的软元件之后。



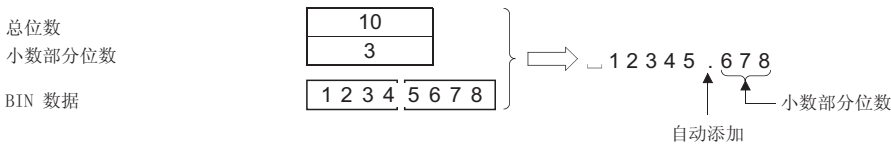
(2) ⑤1能够指定的总位数在 2 到 13 之间。

(3) 能够被⑤1+1 指定为小数的一部分的位数是 0 到 10。
但是，小数点后面的位数必须小于或等于总位数减 3。

(4) ⑤1和⑤2+1 可以指定的 BIN 数据在-2147483648 到 2147483647 之间。

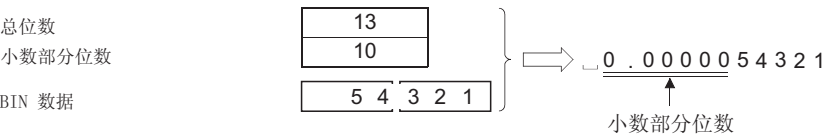
(5) 转换之后的字符串数据按如下所示存储在软元件中：

- ① 如果 BIN 数据为正，则存储符号“20_H”（空格）；如果为负，则存储符号“2D_H”（负号-）。
- ② 如果小数点后的位数的设置是 0 以外的任何数据，则“2E_H”（-）将被自动存储在指定位数的第一位之前。

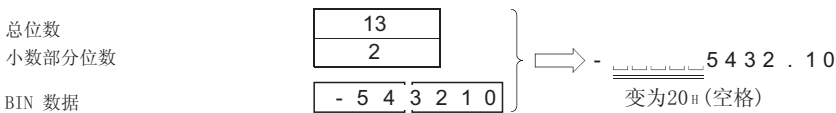


如果小数部分的位数是 0，则该值不存储“2D_H”。

- ③ 如果小数部分比 BIN 数据有更多的位数，将自动增加一个 0，并将该值向右移位，这样转换后的值将变为“0. 0000054321”。



- ④ 如果除了符号和小数点之外的总位数大于 BIN 数据位数，则“20_H”（空格）将被存储在符号和数值之间。



如果 BIN 的位数较大，将返回一个错误。

- ⑤ 值“00_H”被自动存储在已转换字符串的末尾。

[运行错误]

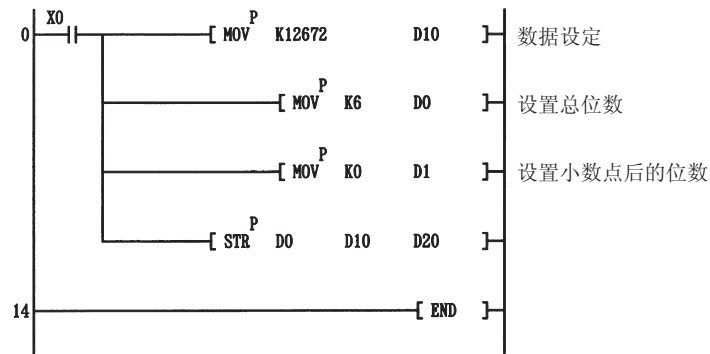
- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 0N，并且错误代码存储在 SD0 上。
- 由③指定的总位数超出了以下所示的范围。 (错误代码: 4100)
当使用 STR 指令时 2 到 8
当使用 DSTR 指令时 2 到 13
- 由③+1 指定的小数部分的位数超出了以下所示的范围。 (错误代码: 4100)
当使用 STR 指令时 0 到 5
当使用 DSTR 指令时 0 到 10

- 由⑤指定的总位数和由⑤+1 指定的小数部分的位数之间的关系不符合以下所示：
(错误代码：4100)
总位数减 3 等于或大于小数部分的位数。
- 由⑤指定的位数加上符号和小数部分的位数小于⑥指定的 BIN 数据的位数。
(错误代码：4100)
- 存储由⑦指定的字符串的软元件范围超出了相应的软元件范围。
(错误代码：4100)

[程序示例]

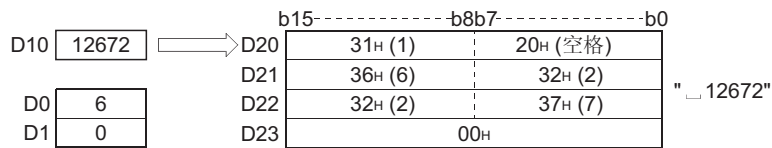
(1) 当 X0 变为 ON 时，以下程序根据 D0 和 D1 的位指定转换存储在 D10 上的 BIN 16 位数据，并将结果存储在 D20 到 D23 上。

[梯形图模式]



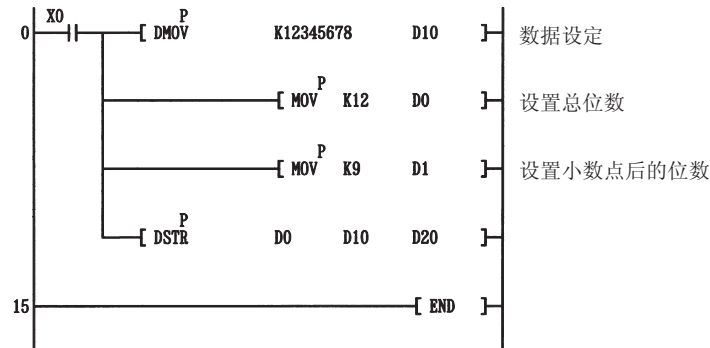
[列表模式]

步	指令	软元件
0	LD	X0
1	MOV ^P	K12672 D10
4	MOV ^P	K6 D0
7	MOV ^P	K0 D1
10	STR ^P	D0 D10
14	END	D20



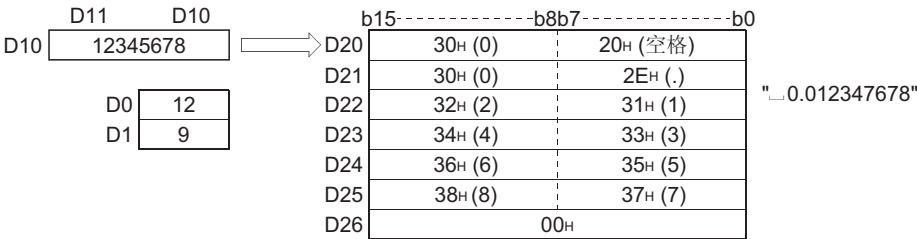
(2) 当 X0 变为 ON 时，以下程序根据 D0 和 D1 上的位指定转换存储在 D10 和 D11 上的 BIN 32 位数据，并将结果存储在 D20 到 D26 上。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	DMOVP	K12345678 D10
5	MOVP	K12 D0
8	MOVP	K9 D1
11	DSTRP	D0 D10 D20
15	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7. 11. 10 从字符串到 BIN 16 位或 32 位数据的转换 (VAL, VALP, DVAL, DVALP)

数据 设定	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		常量 \$
⑤	—	○			—		○
⑥1	○	○			—		—
⑥2	○	○			○		—

[指令符号]	[执行条件]	□ 表示 VAL/DVAL			
VAL, DVAL		命令		⑤	⑥1 ⑥2
VALP, DVALP		命令		⑤	⑥1 ⑥2

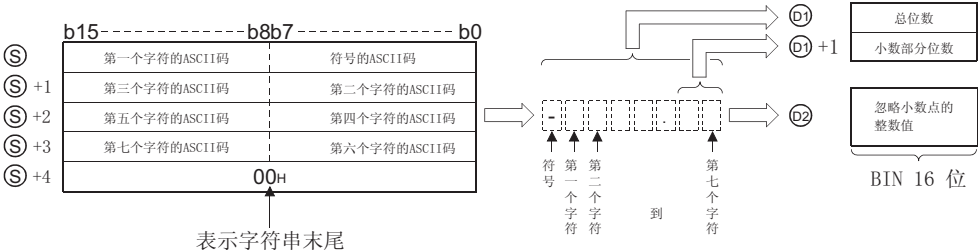
[设定数据]

设定数据	含义	数据类型
⑤	存储即将转换为 BIN 数据的字符串的软元件的起始号	字符串
⑥1	存储转换后的 BIN 数据位数的软元件的起始号	BIN 16 位
⑥2	存储转换后的 BIN 数据的软元件的起始号	BIN 16/32 位

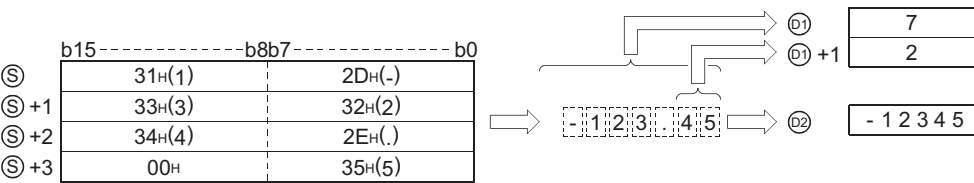
[功能]

VAL

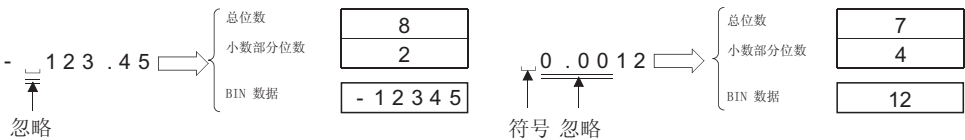
- (1) 将存储在由⑤指定号码的软元件开始的软元件上的字符串转换成 BIN 16 位数据，并将位数和 BIN 数据存储在⑥1和⑥2中。
- 对于字符串到 BIN 的转换，从由⑤指定号码的软元件开始到存储“00H”的软元件点之间的所有数据将被作为字符串处理。



例如，如果⑤之后的软元件指定了字符串“-123.45”，则运算结果将按照以下方式存储在⑥①和⑥②上：

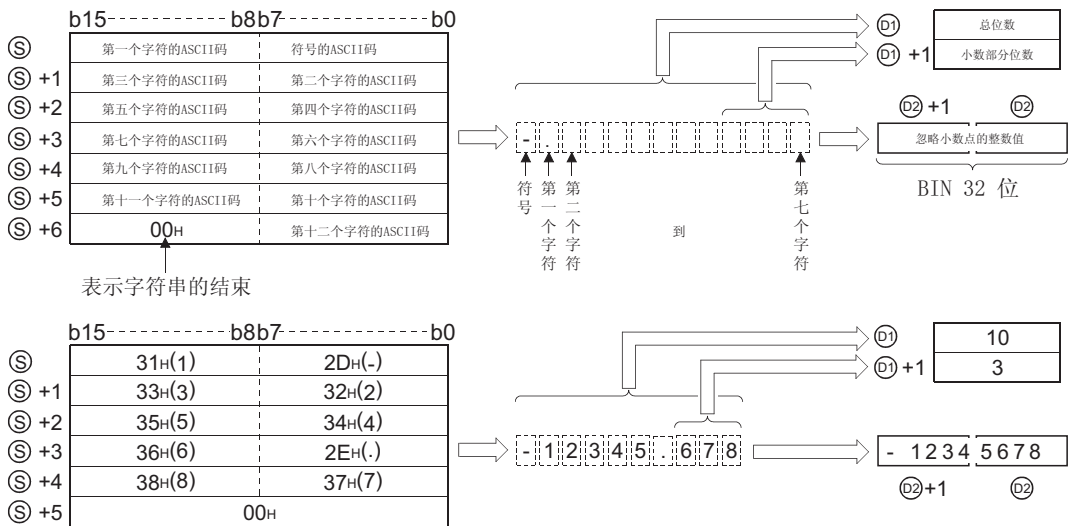


- (2) 可在⑤上指定的字符串的总字符数在 2 到 8 之间。
- (3) 从⑤指定的字符串开始的 0 到 5 个字符可成为小数部分。
但是，该数值必须不超过总位数减 3。
- (4) 当忽略小数点时，可转换成 BIN 值的字符串的数值为-32768 到 32767。
数值字符串，除符号和小数点之外，仅可在“30H”到“39H”的范围内指定。
当忽略小数点时，该值表示如下：
(示例)“-12345.6”被表示为“-123456”。
- (5) 如果数值为正，则存储符号“20H”；如果为负，则存储符号“2DH”。
- (6) 为小数点设置“2EH”。
- (7) ⑥②上存储的总位数是表示数值(包括符号和小数点)的所有字符的总和。
在⑥①+1 上存储的小数点之后的字符包括从“2EH”(.)往前的字符数。
在⑥②上存储的 BIN 数据是已被转换成 BIN 数据、忽略小数点的字符串。
- (8) 如果由⑤指定的字符串中在符号和第一个非零的数值之间有一个“20H”(空格)或者一个“30H”(0)，则在将其转换到 BIN 值期间忽略“20H”或“30H”。



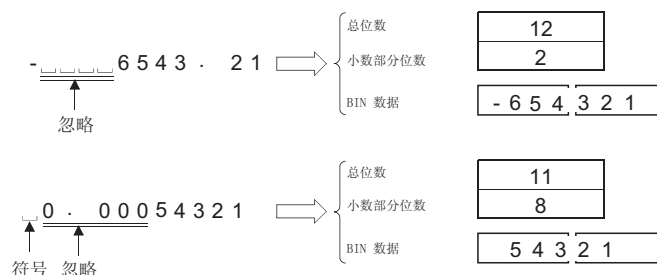
DVAL

- (1) 将存储在由⑤指定号码的软元件之后的字符串转换成 BIN 32 位数据，并存储在①和②上。
对于从字符串到 BIN 的转换，从由⑤指定号码的软元件开始到存储“00H”的软元件点之间的所有数据将被作为字符串处理。



- (2) 由⑤指定的字符串的总字符数在 2 到 13 之间。
- (3) 由⑤指定的字符串的 0 到 10 个字符可成为小数部分。
但是，该数值必须不超过总位数减 3。
- (4) 当忽略小数点时，可转换成 BIN 值的数值字符串是那些落在从 -2147483648 到 2147483647 范围内的字符。
数值字符串，除符号和小数点之外，仅可在从“30H”到“39H”的范围内指定。
- (5) 如果数值为正，则存储符号“20H”；如果为负，则存储符号“2DH”。
- (6) 为小数点设定“2EH”。
- (7) 在①上存储的总位数是表示数值(包括符号和小数点)的所有字符的总和。
在①+1 上存储的小数点之后的字符包括从“2EH”(.) 往前的字符数。
在②上存储的 BIN 数据是已被转换成 BIN 数据、忽略小数点的字符串。

- (8) 如果由⑤指定的字符串中，在符号和第一个非零的数值之间有一个“20_H”（空格）或者一个“30_H”（0），则在将其转换到 BIN 值期间忽略“20_H”或“30_H”。

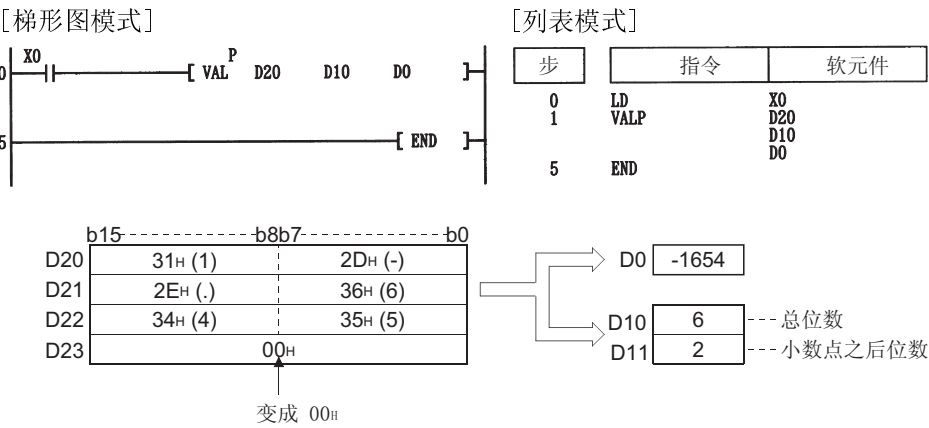


[运行错误]

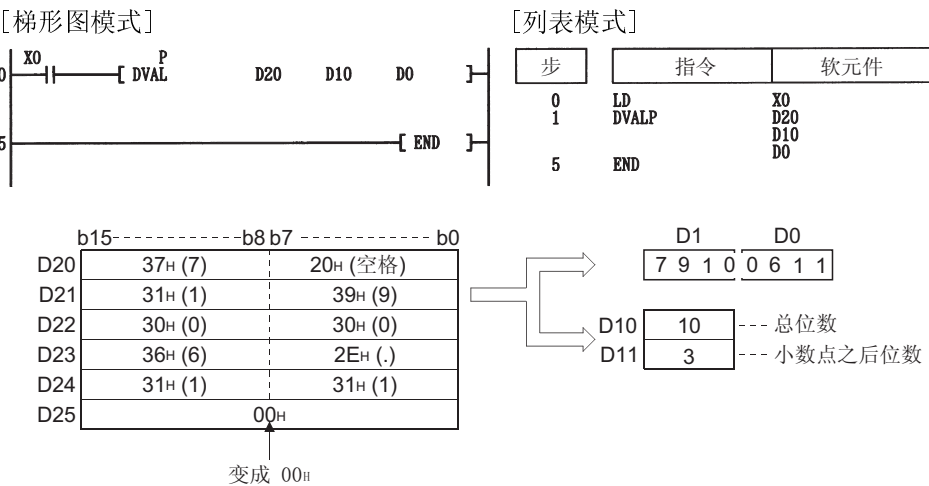
- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 上。
- 由⑤指定的字符串的字符数超出了以下所示的范围：（错误代码：4100）
 - 当使用 VAL 指令时 2 到 8 个字符
 - 当使用 DVAL 指令时 2 到 13 个字符
 - 由⑤指定的字符串的小数部分的字符数超出了以下所示的范围：（错误代码：4100）
 - 当使用 VAL 指令时 0 到 5
 - 当使用 DVAL 指令时 0 到 10
 - 由⑤指定的字符串的总字符数和小数部分的字符数之间的关系超出了以下所示的范围：（错误代码：4100）
 - 总字符数减 3 等于或大于小数部分的字符数
 - 已经为符号设定了一个非“20_H”或“2D_H”的 ASCII 码。（错误代码：4100）
 - 已经将一个非“30_H”到“39_H”或“2E_H”（小数点）的 ASCII 码设定给某个数的一位。（错误代码：4100）
 - 数值中设置的小数点不止一个。（错误代码：4100）
 - 转换后的 BIN 数据值超出了以下范围：（错误代码：4100）
 - 当使用 VAL 指令时 -32768 到 32767
 - 当使用 DVAL 指令时 -2147483648 到 2147483647
 - ASCII 码“00_H”已经被设置在由⑤指定号码的软元件和相关软元件范围的最后软元件号之间的某个地方。（错误代码：4101）

[程序示例]

(1) 当 X0 为 ON 时，以下程序将存储在 D20 到 D22 上的字符串数据当作一个整数读取，并将其转换成一个 BIN 值，存储在 D0 上。



(2) 当 X0 为 ON 时，以下程序将 D20 到 D24 上的字符串数据当作一个整数读取，将其转换成一个 BIN 值，并存储在 D0 上。

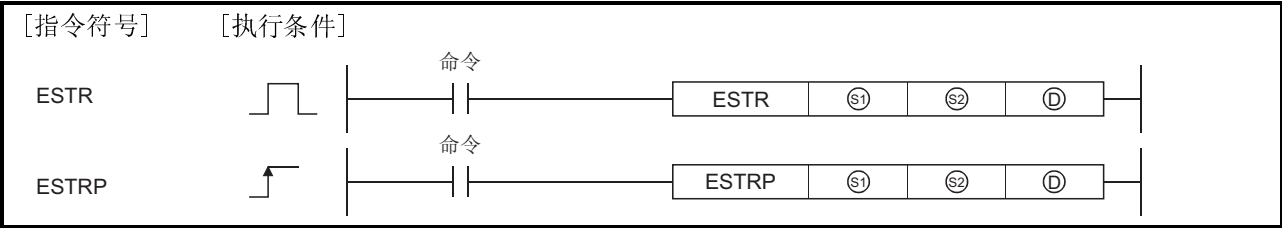


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

※：系列号前五位是04122或更大的。

7. 11. 11 从浮动小数点到字符串数据的转换 (ESTR, ESTRP)

数据设定	可用软元件						
	内部软元件(系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
①	—	○		—	○		—
②	—	○		—	—		—
③	—	○		—	—		—



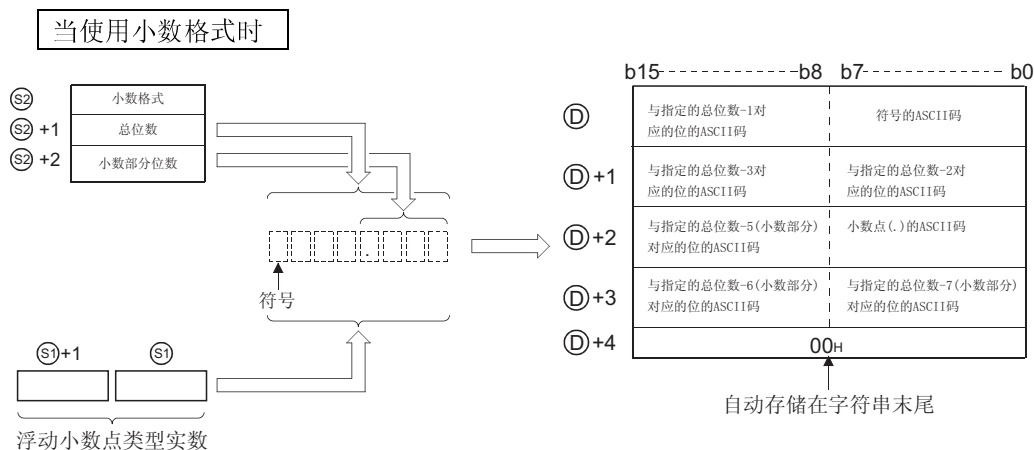
[设定数据]

设定数据	含义	数据类型
①	即将转换的浮点数，或者存储这类数据的软元件的起始号	实数
②	存储将要转换的数值的显示指定的软元件的起始软元件号	BIN 16 位
③	存储已转换字符串的软元件的起始软元件号	字符串

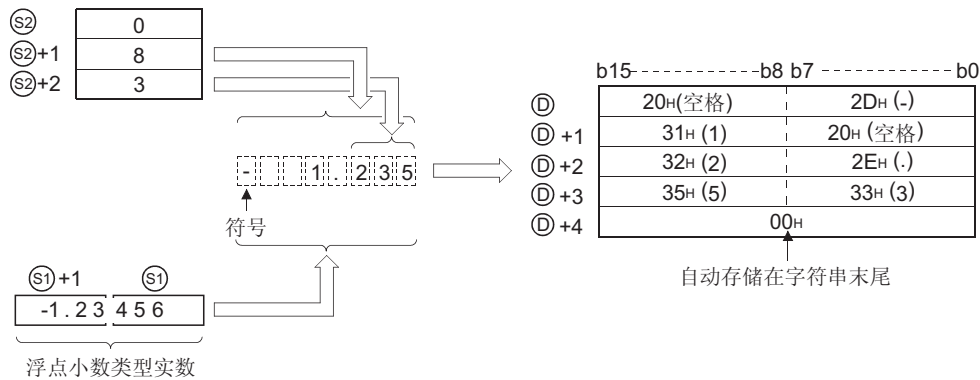
[功能]

- (1) 根据存储在由②指定的软元件之后的显示指定，将由①指定的软元件上存储的浮点型实数转换成字符串，并将结果存储在由③指定号码的软元件之后。
- (2) 转换后的数据将依据由②指定的显示指定而有所不同。

②	0: 小数格式 1: 指数格式	转换后的数据依据在②上选择的格式而有所不同
②+1	总位数	
②+2	小数部分位数	--- 可以设定为2到24



例如，如果共有 8 位，其中 3 位来自小数部分，且指定的值为-1.23456，则运算结果将被按照以下方式存储在④之后。



(a) ⑤2+1 可以指定的总位数显示如下：

当小数点之后的位数是“0”时

..... 总位数 ≥ 2

当小数点之后的位数是某个非“0”值时

..... 总位数 \geq (小数点之后的位数+3)

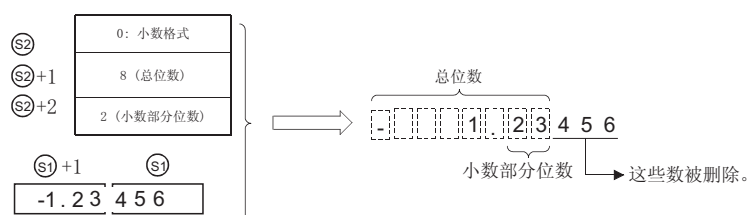
(b) 小数点后可指定的位数在 0 到 7 之间。

但是，小数点后的位数应该等于或小于总位数减 3。

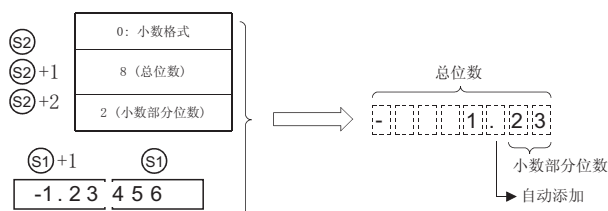
(c) 转换后，按如下所示将字符串数据存储在④之后的软元件上：

① 如果浮点型实数为正，则存储符号“20H”(空格)；如果为负，则存储符号“2DH”(负号-)。

- ② 如果在位数范围内有浮动小数点实数跟在小数点之后，并且十进制值还未被存储，则将删除低位的十进制值。

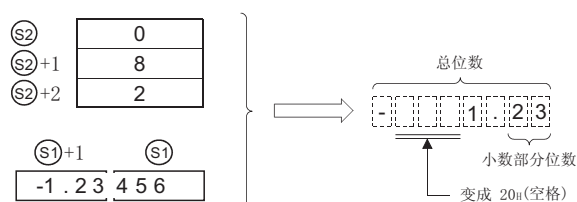


- ③ 如果小数点后的位数被设置成 0 以外的其他值，则“2EH” (.) 将被自动存储在指定位数的第一位之前的位置。

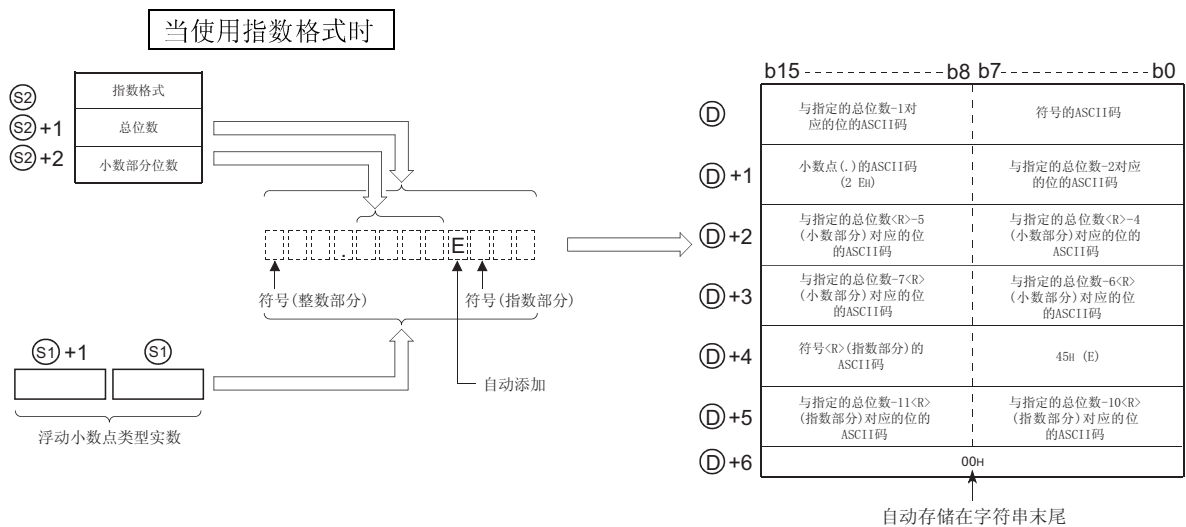


如果该数的小数部分位数为“0”，则不存储 ASCII 码“2EH” (.)

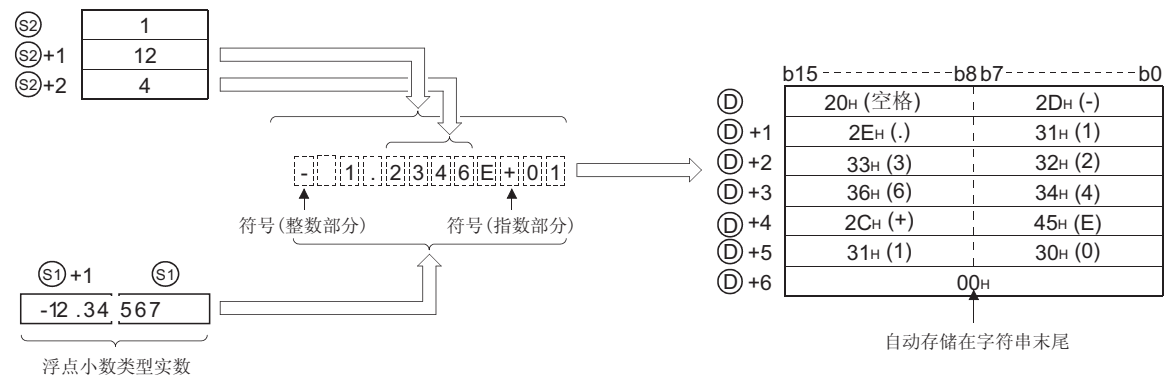
- ④ 如果总位数，不包括符号、小数点和小数点之后的数，大于浮点型实数的小数点之前的部分，则将在符号和该数据的小数点之前的部分之间存储“20H (空格)”。



- ⑤ 在已转换的字符串的末尾自动存储值“00H”。



例如，如果总共有 12 位，其中小数部分有 4 位，且指定值是-12.34567，则运算结果将按照以下方式存储在①之后。

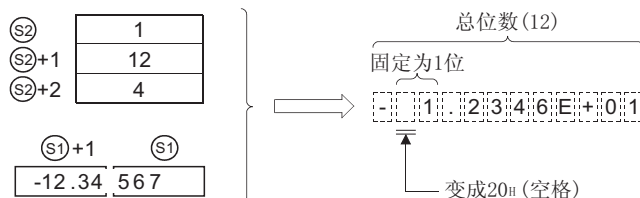


- (a) ②+1 可指定的总位数显示如下：
- 当小数点后的位数为“0”时
- 总位数 ≥ 2
- 当小数点后的位数是某个非“0”值时
- 总位数 ≥ (小数部分位数+7)
- (b) 可指定的跟在小数点之后的位数在 0 到 7 之间。
- 但是，小数部分的位数应等于或小于总位数减 7。

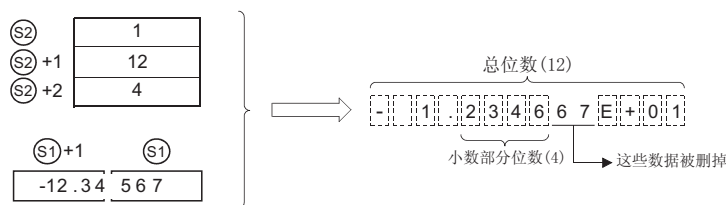
(c) 转换后，按照以下的指示将字符串数据存储在⑤之后的软元件上：

- ① 如果浮点型实数的值为正，则整数前的符号将被存储为 ASCII 码 “20_H” (空格)；如果是一个负值，则符号将被存储为 “2D_H” (-)。
- ② 整数部分被固定为一位。

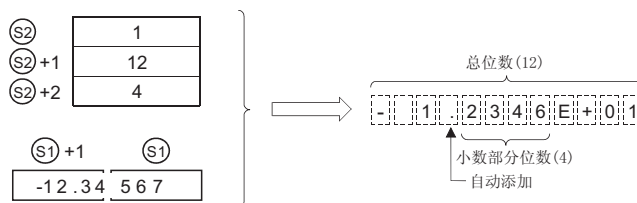
“20_H” (空格) 将被存储在整数和符号之间。



- ③ 如果浮点型实数太长，超出了总位数能容纳的范围，则小数部分的尾数将从结果中删除。



- ④ 如果小数点之后的位数已被设定为任何非 “0” 的值，则 “2E_H” (.) 将被自动存储在指定位数第一位之前的位置。

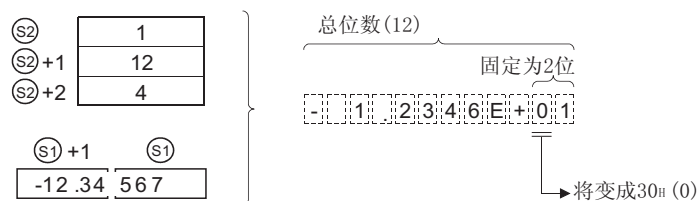


如果该数的小数部分位数为 “0”，则将不存储 ASCII 码 “2E_H” (.)。

- ⑤ 如果指数值为正，则将存储 ASCII 码 “2C_H” (+) 作为该值指数部分的符号；如果指数是一个负值，则将存储 “2D_H” (-)。

⑥ 指数部分固定为 2 位。

如果指数部分仅 1 位，则将在该数的符号和指数之间存储 ASCII 码 “30H” (0)。



⑦ 自动将值 “00H” 存储在已转换字符串的末尾处。

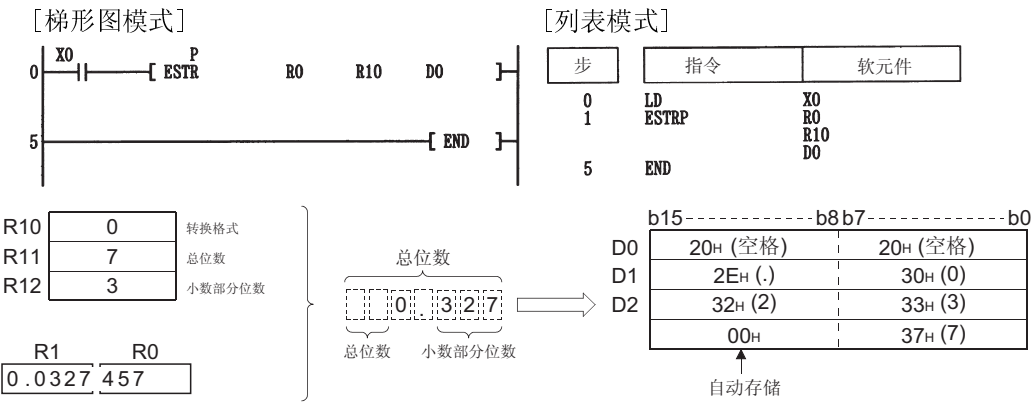
[运行错误]

(1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON。

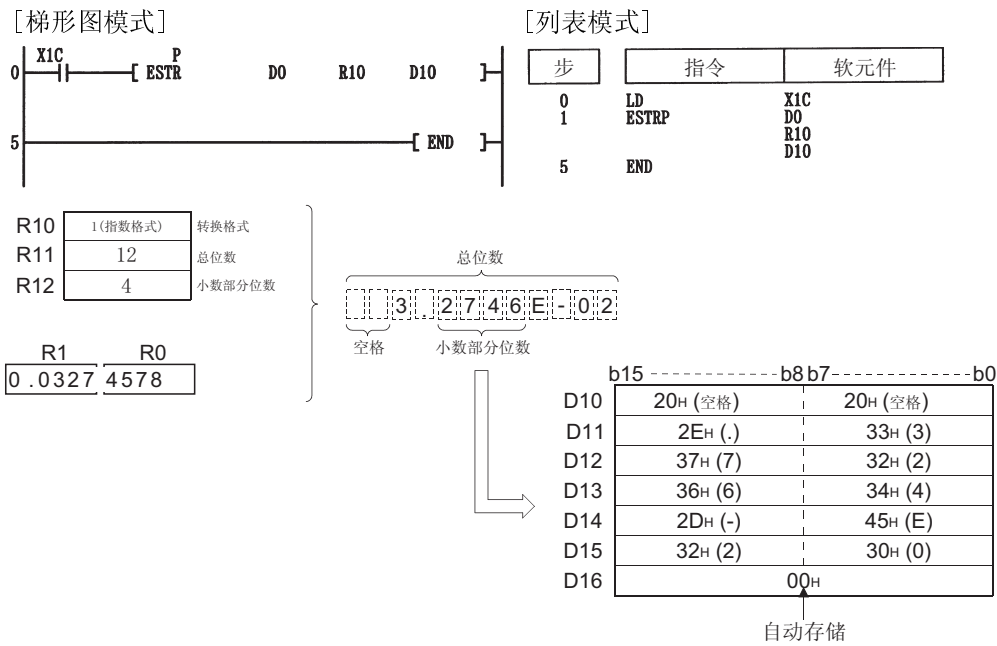
- ⑤1 值不在以下所示的范围内： (错误代码：4100)
 $0, \pm 2^{-126} \leq \text{⑤1} < \pm 2^{128}$
- ⑤2 指定的格式既不是 0 也不是 1。 (错误代码：4100)
- ⑤2+1 指定的总位数超出了以下所示的范围： (错误代码：4100)
 - 当使用小数格式时
 - 当小数点后的位数为 “0” 时
 总位数 ≥ 2
 - 当小数点后的位数是某个非 “0” 值时
 总位数 $\geq (\text{小数部分位数} + 3)$
 - 当使用指数格式时
 - 当小数点后的位数为 “0” 时
 总位数 ≥ 2
 - 当小数点后的位数是某个非 “0” 值时
 总位数 $\geq (\text{小数部分位数} + 7)$
- 给 ⑤2+2 指定值的小数部分指定的位数超出了以下所示的范围： (错误代码：4100)
 - 当使用小数格式时
 小数部分位数小于或等于总位数减 3。
 - 当使用指数格式时
 小数部分位数小于或等于总位数减 7。
- 存储由 ⑤指定的字符串的软元件范围超出了相应的软元件范围。 (错误代码：4101)

[程序示例]

(1) 当 X0 变 ON 时，以下程序根据 R10 到 R12 上存储的转换指定，转换存储在 R0 和 R1 上的浮点型实数，并将结果存储在 D0 之后。



(2) 当 X1C 变 ON 时，以下程序根据 R10 到 R12 上存储的转换指定，转换存储在 D0 和 D1 上的浮点型实数，并将结果存储在 D0 之后。

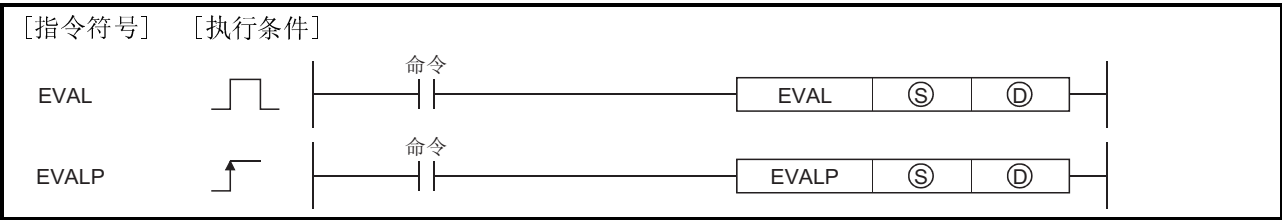


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

※：系列号前五位是04122或更大的。

7. 11. 12 从字符串到浮点数的转换 (EVAL, EVALP)

数据 设定	可用软元件								
	内部软元件(系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\C[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 \$	其它
	位	字		位	字				
⑤	—	○		—	—	—	—	○	—
⑥	—	○		—	○	○	—	—	—

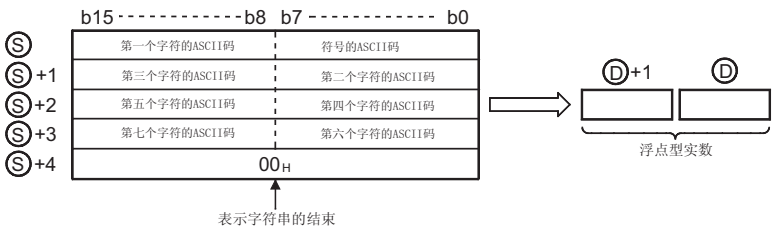


[设定数据]

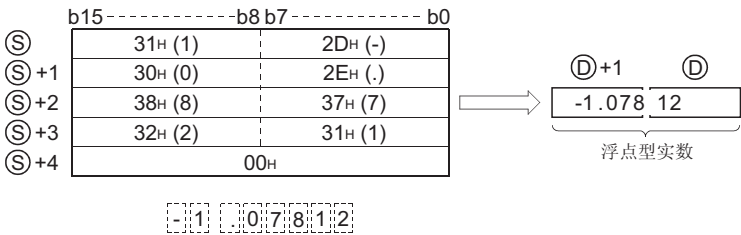
设定数据	含义	数据类型
Ⓔ	即将转换为浮点型实数的字符串数据，或存储该数据的软元件的起始软元件号	字符串
Ⓕ	转换后存储浮点型实数的软元件的起始软元件号	实数

[功能]

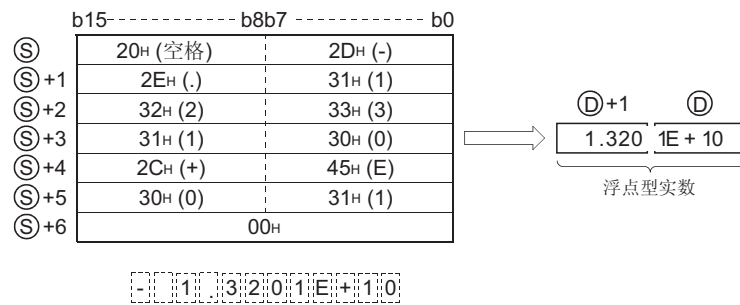
- (1) 将存储在由Ⓔ指定号码的软元件之后的字符串转换成浮点型实数，并将结果存储在Ⓕ指定的软元件上。
- (2) 指定的字符串能够以小数格式或者指数格式转换成浮点型实数。



(a) 当使用小数格式时

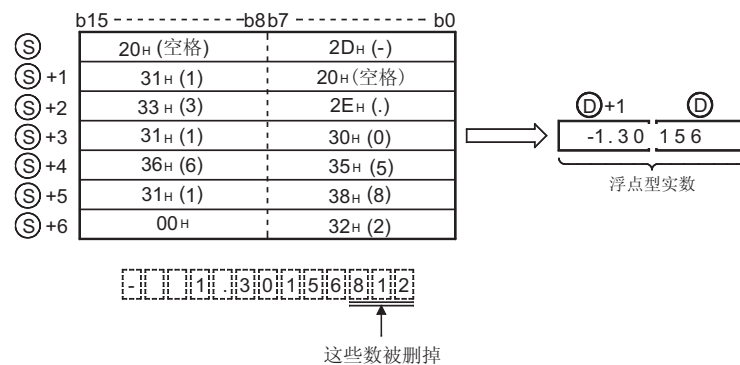


(b) 当使用指数格式时

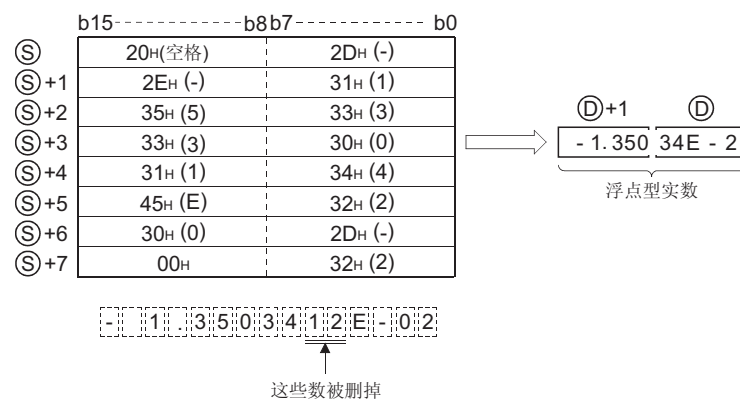


(3) 除了结果的符号、小数点和指数部分之外，由⑤指定的即将转换成浮点型实数的字符串的 6 位将是有效的，自第 7 位开始的所有数据将从结果中删除。

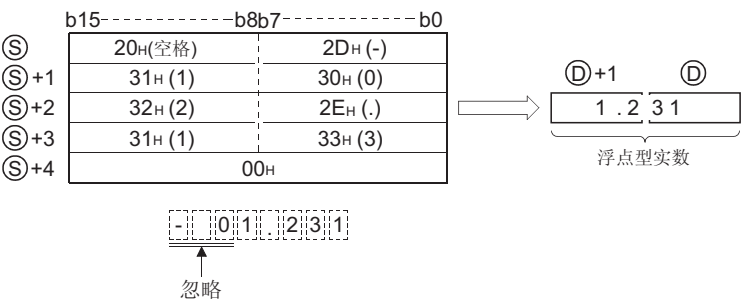
(a) 当使用小数格式时



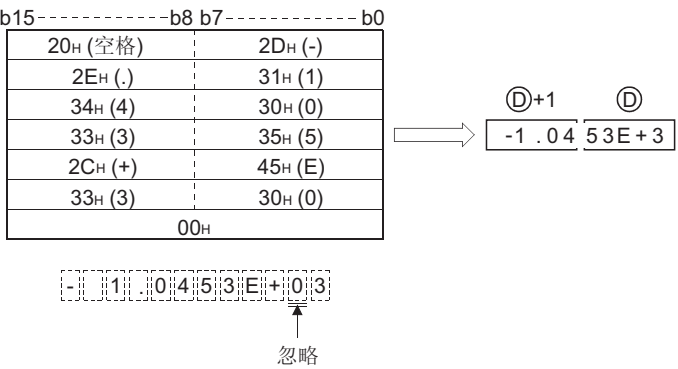
(b) 当使用指数格式时



(4) 如果，在⑤指定的字符串中，除起始的零之外，有 ASCII 码“20_H”（空格）或“30_H”（0），则在转换时将忽略它们。



(5) 如果在一个指数格式的字符串中，在字符“E”和一个数之间存在 ASCII 码“30_H”（0），则当进行转换时将忽略“30_H”。



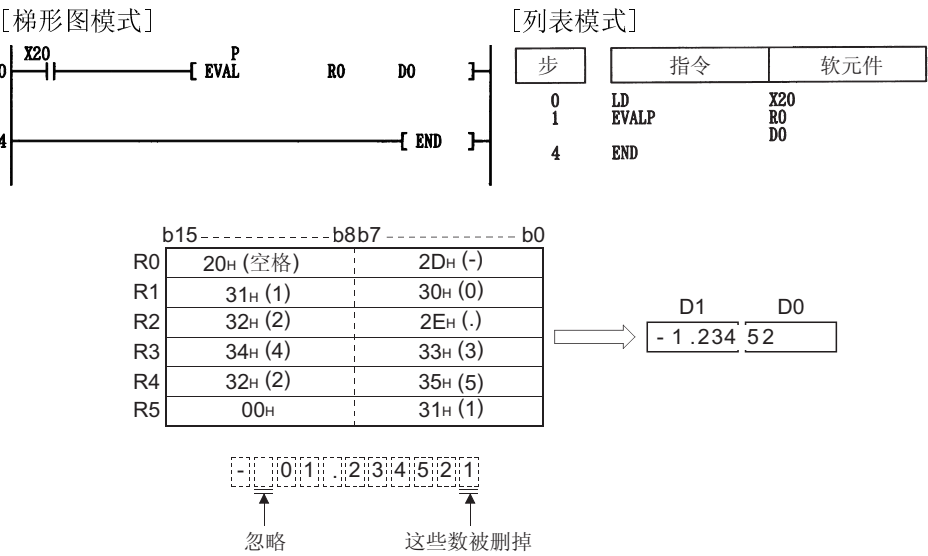
(6) 一个字符串最多可设定 24 个字符。

[运行错误]

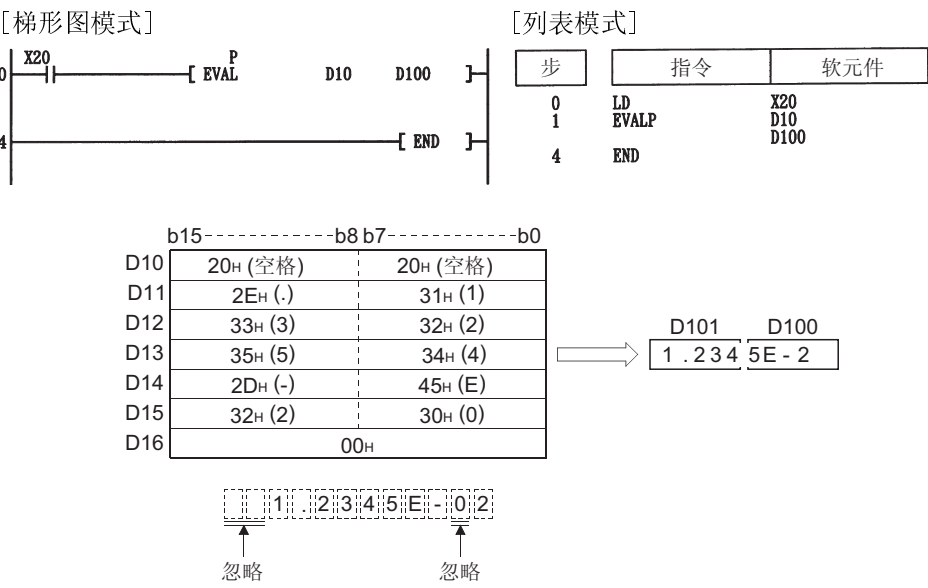
- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 0N，并且错误代码存储在 SDO 上。
- 整数部分或者小数部分含有一个不在“30_H”（0）到“39_H”（9）范围内的字符。(错误代码: 4100)
 - 字符“2E_H”（.）在由⑤指定的字符串中出现了两次或两次以上。(错误代码: 4100)
 - 该值的指数部分包含有“45_H”（E），“2C_H”（+）或“45_H”（E），“2D_H”（-）之外的字符，或者有不只一个指数。(错误代码: 4100)
 - 转换后的数据不在以下范围内：(错误代码: 4100)
 $0, \pm 2^{-126} \leq |\text{转换后的数据}| < \pm 2^{128}$
 - “00_H”代码没有在⑤到相关软元件的范围内出现。(错误代码: 4101)
 - 字符串中的字符数是 0 或大于 24。(错误代码: 4100)

[程序示例]

(1) 当 X20 变为 ON 时，以下程序将存储在 R0 之后的字符串转换成浮点型实数，并将结果存储在 D0 和 D1 上。




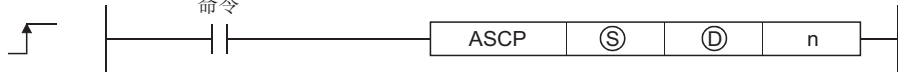
(2) 当 X20 变为 ON 时，以下程序将存储在 D10 之后的字符串转换成浮点型实数，并将结果存储在 D100 和 D101 上。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7. 11. 13 从十六进制 BIN 到 ASCII 的转换(ASC, ASCP)

数据 设定	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
Ⓔ	—	○			—				—
Ⓕ	—	○			—				—
n	○	○			○				—

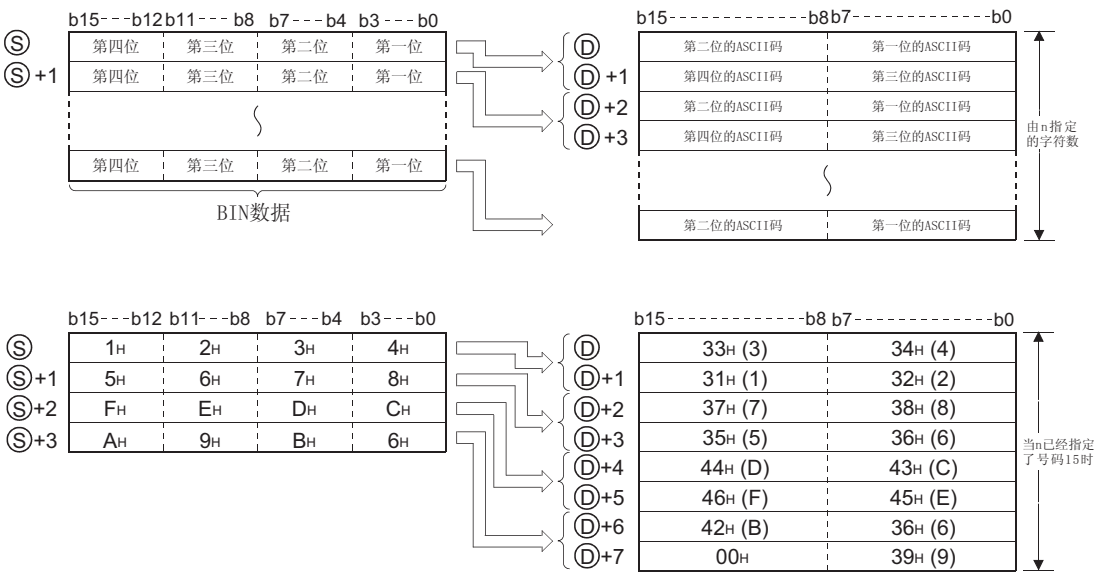
[指令符号]	[执行条件]
ASC	
ASCP	

[设定数据]

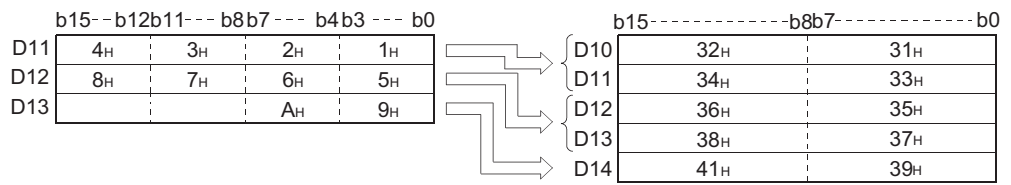
设定数据	含义	数据类型
Ⓢ	存储即将转换成字符串的 BIN 数据的软元件的起始软元件号	BIN 16 位
Ⓓ	存储已转换的字符串的软元件的起始软元件号	字符串
n	存储的字符数	BIN 16 位

[功能]

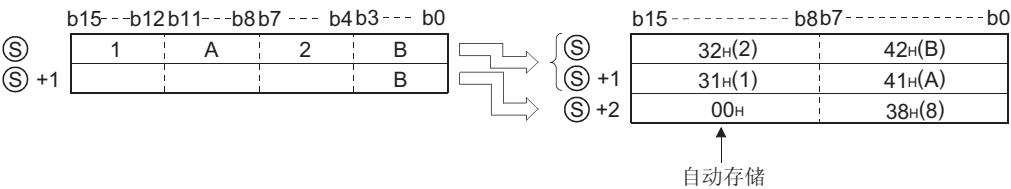
(1) 将存储在由Ⓢ指定号码的软元件之后的 BIN 16 位数据转换成十六进制 ASCII，并将在由 n 指定的字符数的范围内的结果存储在由Ⓓ指定号码的软元件之后。



- (2) 使用 n 设定字符数，使得由⑤指定的 BIN 数据的范围和由⑥指定的字符串存储软元件的范围被自动设定。
- (3) 即使存储即将转换的 BIN 数据的软元件范围和存储转换后的 ASCII 数据的软元件范围有重叠，仍将准确地进行处理。



- (4) 如果 n 已经指定了奇数个字符，则 ASCII 码“00H”将被自动存储在字符串存储范围内的最后一个软元件的头 8 位中。



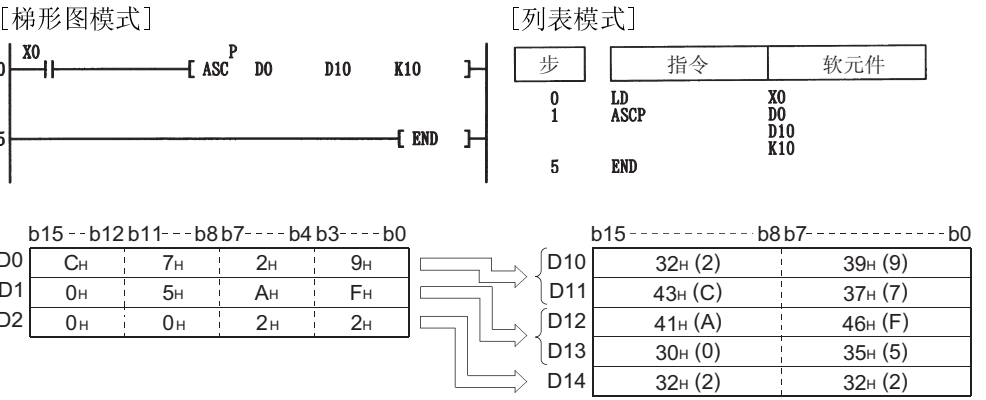
- (5) 如果 n 指定的字符数是“0”，则将不进行转换处理。

[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 上。
- 在由⑤指定号码的软元件之后，由 n 指定的字符数的范围超出了相关软元件的范围。
(错误代码：4101)
 - 在由⑥指定号码的软元件之后，由 n 指定的字符数的范围超出了相关软元件的范围。
(错误代码：4101)

[程序示例]

- (1) 当 X0 变为 ON 时，以下程序将存储在 D0 上的 BIN 数据作为十六进制值进行读取，将其转换成字符串，并将结果存储在 D10 到 D14 上。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7. 11. 14 从 ASCII 到十六进制 BIN 的转换 (HEX, HEXP)

数据 设定	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○				—	
⑥	—	○				—	
n	○	○				○	

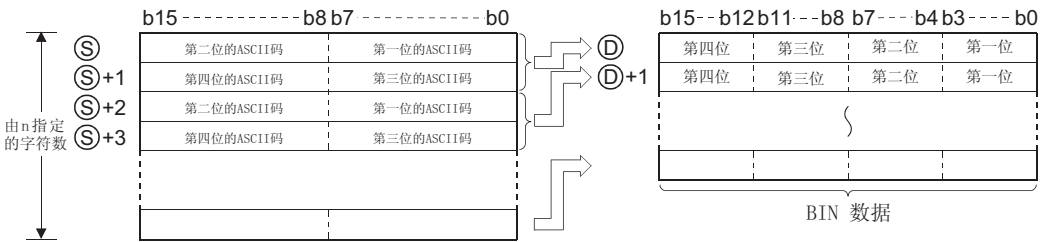
[指令符号]	[执行条件]
HEX	
HEXP	

[设定数据]

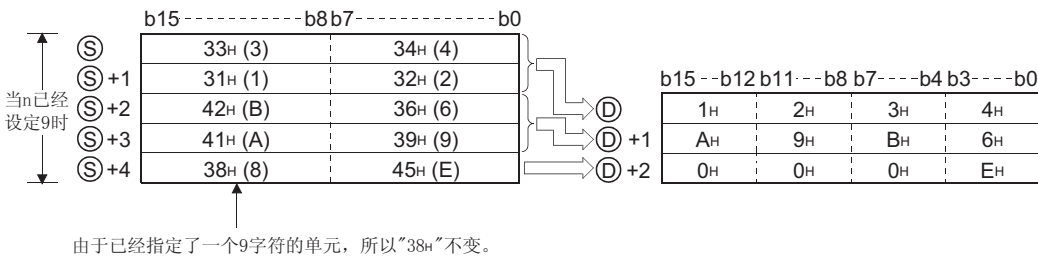
设定数据	含义	数据类型
⑤	存储即将转换成 BIN 数据的字符串的软元件的起始软元件号	字符串
⑥	存储已转换的 BIN 数据的软元件的起始软元件号	BIN 16 位
n	转换的字符数	BIN 16 位

[功能]

(1) 将存储在由⑤指定号码的软元件之后的 BIN 16 位数据转换成十六进制 ASCII，并将在由 n 指定的字符数的范围内的结果存储在由⑥指定号码的软元件之后。

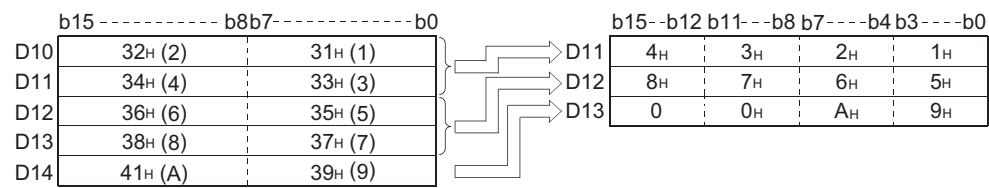


例如，如果 n 指定了数字 9，操作将如下所示：

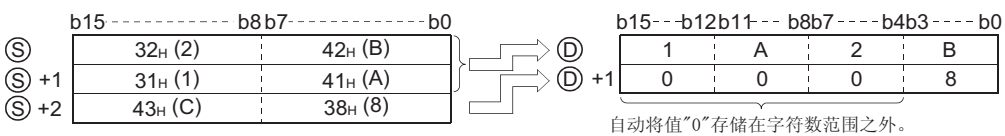


(2) 使用 n 设定字符数，使得由⑤指定的字符串的范围和存储由⑥指定的 BIN 数据的软元件的范围被自动设定。

(3) 即使存储即将转换的 ASCII 码的软元件范围和存储已转换的 BIN 数据的软元件范围有重叠，仍将进行准确处理。



(4) 如果 n 指定的字符数不能被 4 整除，则将自动将“0”存储在存储已转换 BIN 值的软元件的最后一个软元件号中指定的字符数之后。



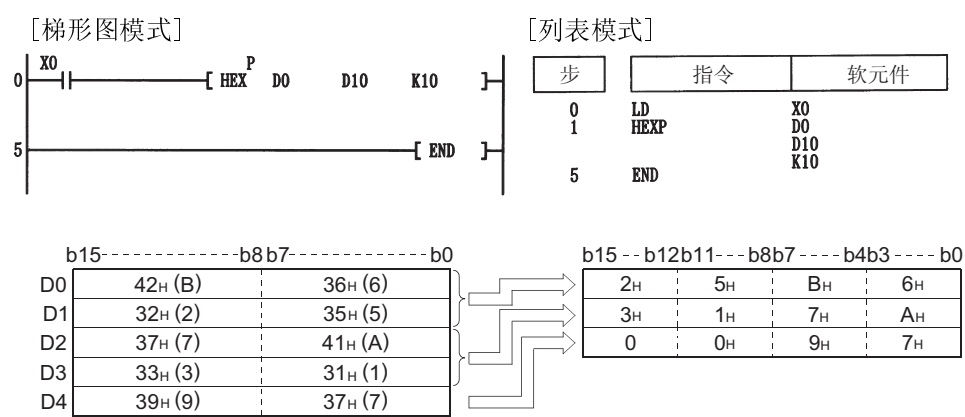
- (5) 如果 n 指定的字符数为“0”，则将进行不转换处理。
- (6) Ⓢ可指定的 ASCII 码包括“30_H”到“39_H”和“41_H”到“46_H”。

[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 上。
- Ⓢ设定了十六进制字符串之外的字符 (即这些字符不在“30_H”到“39_H”或“41_H”到“46_H”的范围内)。 (错误代码: 4100)
 - 在由Ⓢ指定号码的软元件之后，由 n 指定的字符数的范围超出了相关软元件范围。 (错误代码: 4101)
 - 在由Ⓣ指定号码的软元件之后，由 n 指定的字符数的范围超出了相关软元件范围。 (错误代码: 4101)
 - n 指定的字符数是一个负值。

[程序示例]

(1) 当 X0 变为 ON 时，以下程序将存储在 D0 到 D4 上的字符串转换成 BIN 数据，并将结果存储在 D10 到 D14 上。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7. 11. 15 从右边或从左边提取字符串数据 (RIGHT, RIGHTP, LEFT, LEFTP)

数据 设定	可用软元件									
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量		其它
	位	字		位	字			K, H	\$	
⑤	—	○		—				—	○	—
⑥	—	○		—				—	—	—
n	○	○		○				○	—	—

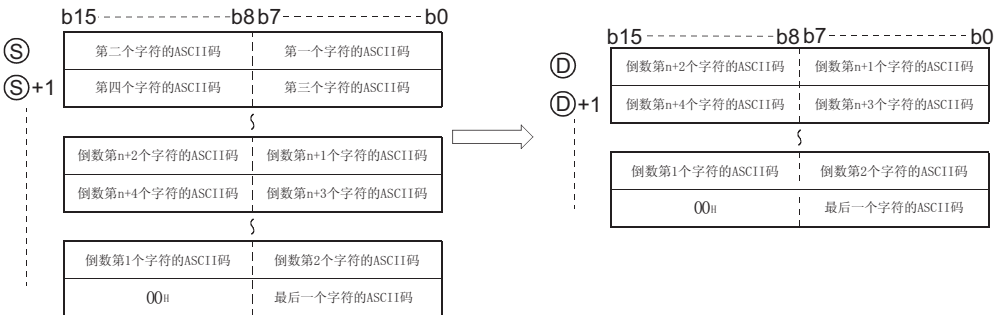
[指令符号]	[执行条件]	□ 表示 RIGHT/LEFT			
RIGHT, LEFT		命令	□	⑤	⑥ n
RIGHTP, LEFTP		命令	□ P	⑤	⑥ n

[设定数据]

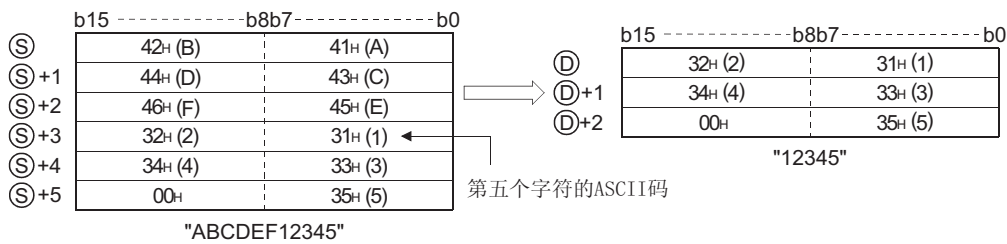
设定数据	含义	数据类型
⑤	存储字符串的软元件的起始软元件号	字符串
⑥	存储⑤右边或左边的 n 个字符的字符串的软元件的起始软元件号	
n	提取的字符数	BIN 16 位

[功能]

- RIGHT
- (1) 将存储在由⑤指定号码的软元件之后的软元件上的字符串右侧 (字符串的末尾) 的 n 个字符, 存储在由⑥指定号码的软元件之后的软元件中。



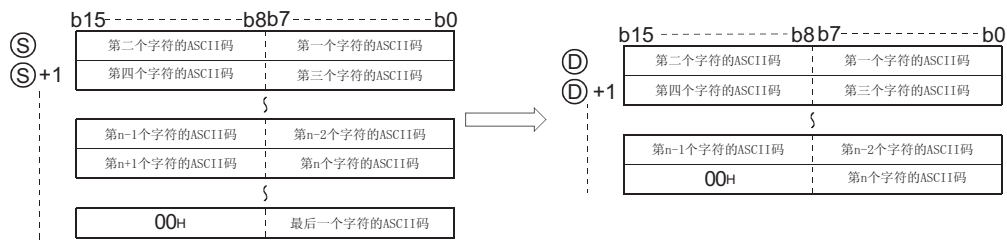
当 n=5 时



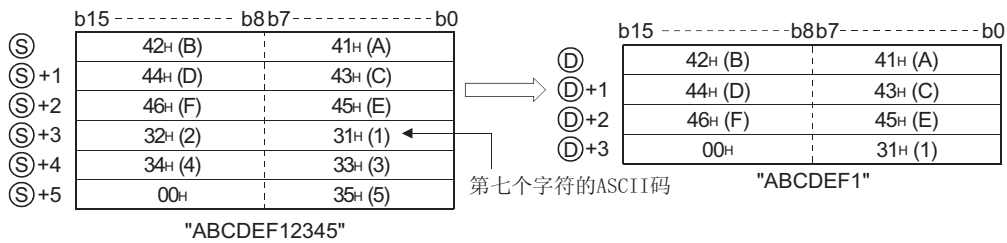
(2) 如果 n 指定的字符数为“0”，则 NULL 代码 (00H) 将被存储在Ⓓ上。

LEFT

(1) 将存储在由Ⓔ指定号码的软元件之后的软元件上的字符串左侧 (字符串的开始) 的 n 个字符，存储在由Ⓓ指定号码的软元件之后的软元件中。



当 n=7 时



(2) 如果 n 指定的字符数为“0”，则 NULL 代码 (00H) 将被存储起来。

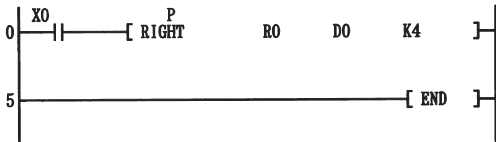
[运行错误]

- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 0N，并且错误代码存储在 SD0 上。
- n 值超过了由Ⓔ指定的字符数。 (错误代码: 4101)
 - 从Ⓓ开始的 n 个字符的范围超出了相关软元件的范围。 (错误代码: 4101)

[程序示例]

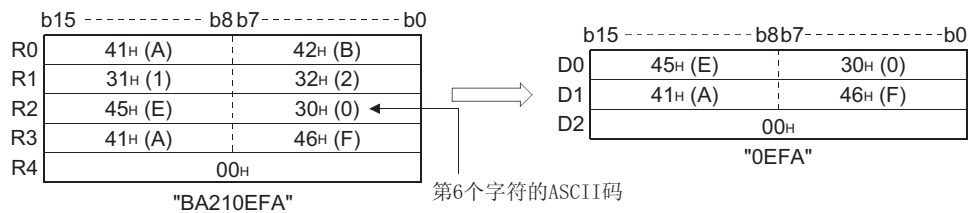
(1) 当 X0 变 ON 时，以下程序将 R0 上存储的字符串右侧的 4 个字符的数据存储在 D0 上。

[梯形图模式]



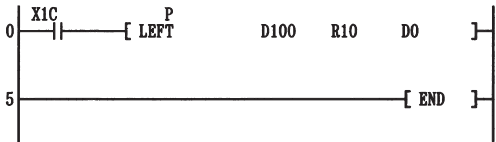
[列表模式]

步	指令	软元件
0	LD	X0
1	RIGHTP	R0 D0 K4
5	END	



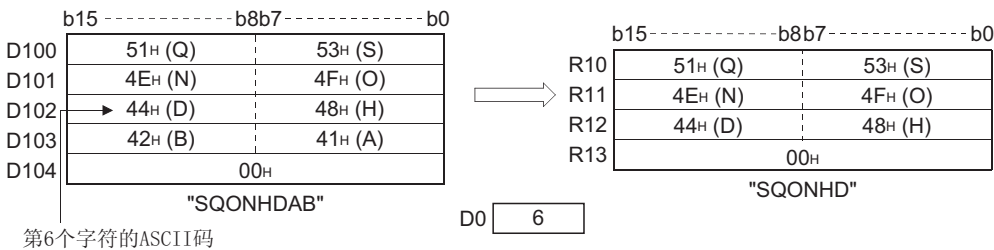
(2) 当 X1C 变为 ON 时，以下程序将 D100 上存储的字符串数据左侧与 D0 中存储的值对应的字符数存储在 R10 之后。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X1C
1	LEFTP	D100 R10 D0
5	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.11.16 字符串的随机选择和替换(MIDR, MIDRP, MIDW, MIDWP)

设定数据	可用的软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [J][N][C]		特殊功能模 块 [U][V][G]	变址寄存器 Zn
	位	字		位	字		
Ⓢ1	—	○		—			○
Ⓓ	—	○		—			—
Ⓢ2	○	○		○			—

[指令符号 1]	[执行条件]	□ 表示 MIDR/MIDW
MIDR, MIDW		
MIDRP, MIDWP		

[设定数据]

设定数据	含义	数据类型
Ⓢ1	存储字符串的软元件的起始软元件号	字符串
Ⓓ	存储运算结果的字符串数据的软元件的起始软元件号	
Ⓢ2	存储起始字符的位置和字符数的软元件的起始软元件号 ● Ⓢ2+1: 起始字符的位置 ● Ⓢ2+1: 字符数	BIN 16 位

[功能]

MIDR

(1) 从由Ⓢ1指定的软元件号开始，从正被存储的字符串的左侧，将从由Ⓢ2指定的位置开始的由Ⓢ2+1指定数目的字符，存储到从由Ⓓ指定的软元件号开始的位置。

Ⓢ1

Ⓢ1+1

Ⓢ1+2

Ⓢ1+3

Ⓢ1+4

Ⓢ1+5

b15-----b8	b7-----b0
42H (B)	41H (A)
44H (D)	43H (C)
46H (F)	45H (E) ←
48H (H)	47H (G)
4AH (J)	49H (I) ←
00H	4BH (K)

"ABCDEFGHIJK"

Ⓢ2

Ⓢ2+1

5	5
---	---

Ⓓ

Ⓓ+1

Ⓓ+2

b15-----b8	b7-----b0
46H (F)	45H (E)
48H (H)	47H (G)
00H	49H (I)

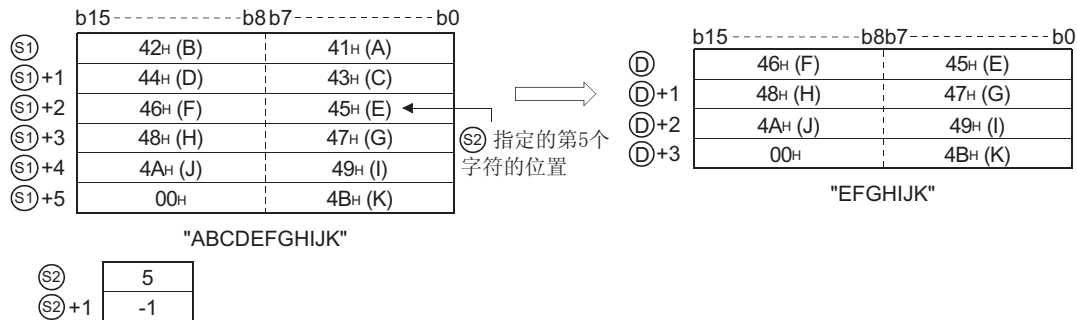
"EFGHI"

Ⓢ2 指定的第5个字符的位置

Ⓢ2+1 指定的第5个字符的ASCII码

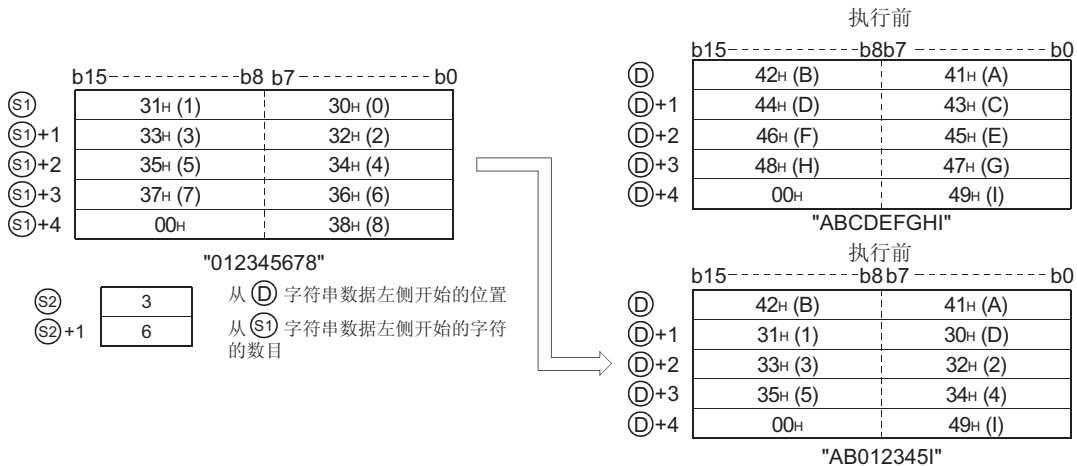
(2) 如果由Ⓢ2+1 指定的字符数是“0”，那么不执行处理。

- (3) 如果由②+1 指定的字符数为“-1”，那么从由④指定的软元件开始存储数据，直到碰到由③指定的最后一个字符。

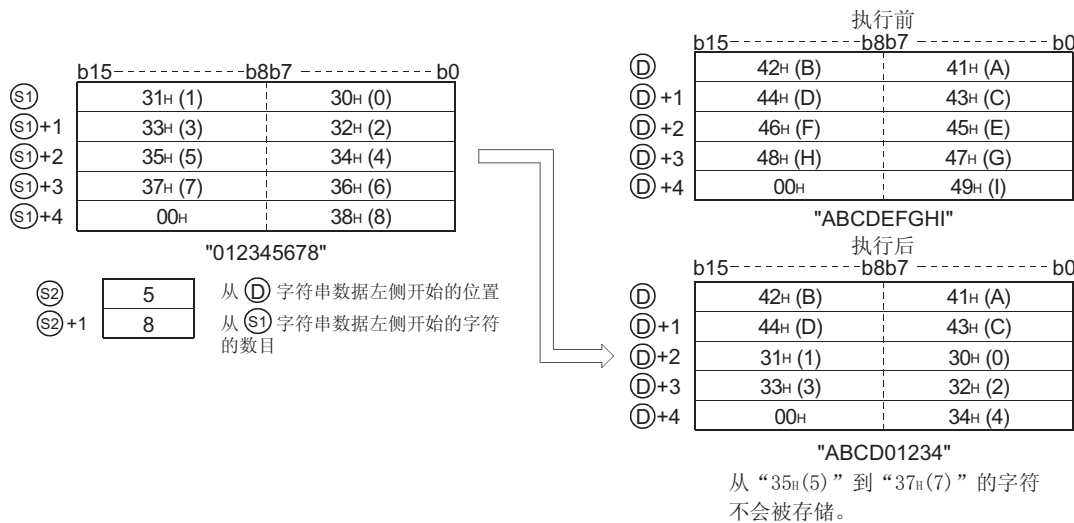


MIDW

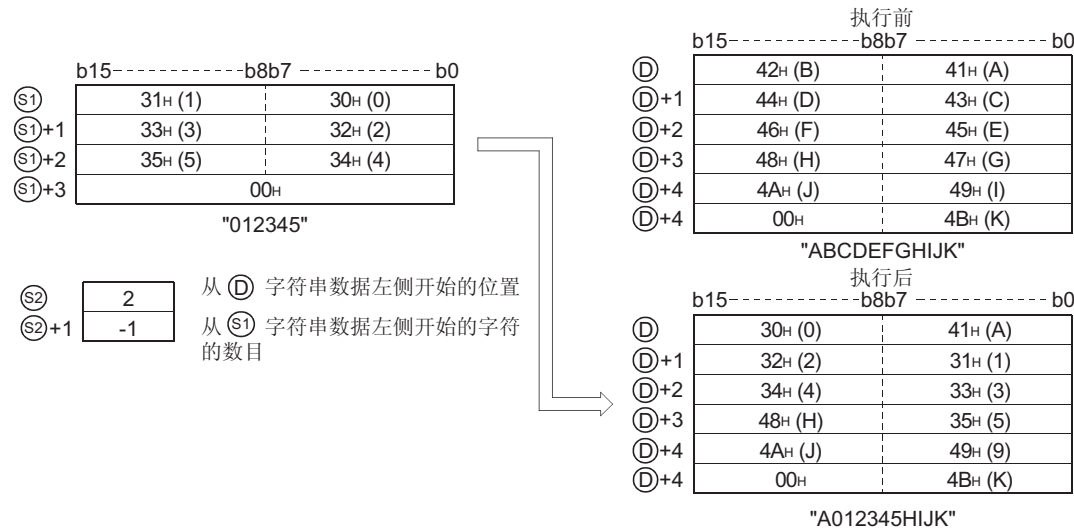
- (1) 从①指定的软元件号开始，从正被存储的字符串数据的左侧开始，将由②+1 指定数目的字符，存储到由④指定的软元件号开始的，从正被存储的字符串数据的左侧，由③指定的位置开始的位置中去。



- (2) 如果由②+1 指定的字符数是 0，那么将不执行处理。
- (3) 如果由②+1 指定的字符数超出了从由④指定的字符串数据开始的最后一个字符，那么存储数据，只到碰到最后一个字符。



- (4) 如果由②+1 指定的字符数为“-1”，那么从由①指定的软元件开始存储字符，直到碰到由③指定的最后一个字符数据。



[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。

对于 MIDR 指令

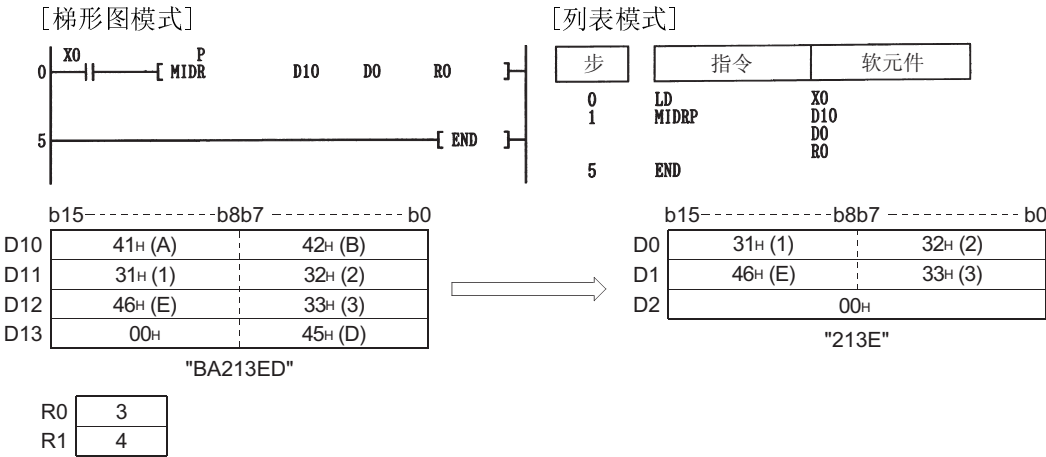
- ②的值超出了由③指定的字符数。(错误代码: 4101)
- 从位置①开始的②+1 个字符数超出了①软元件的范围。(错误代码: 4101)

对于 MIDW 指令

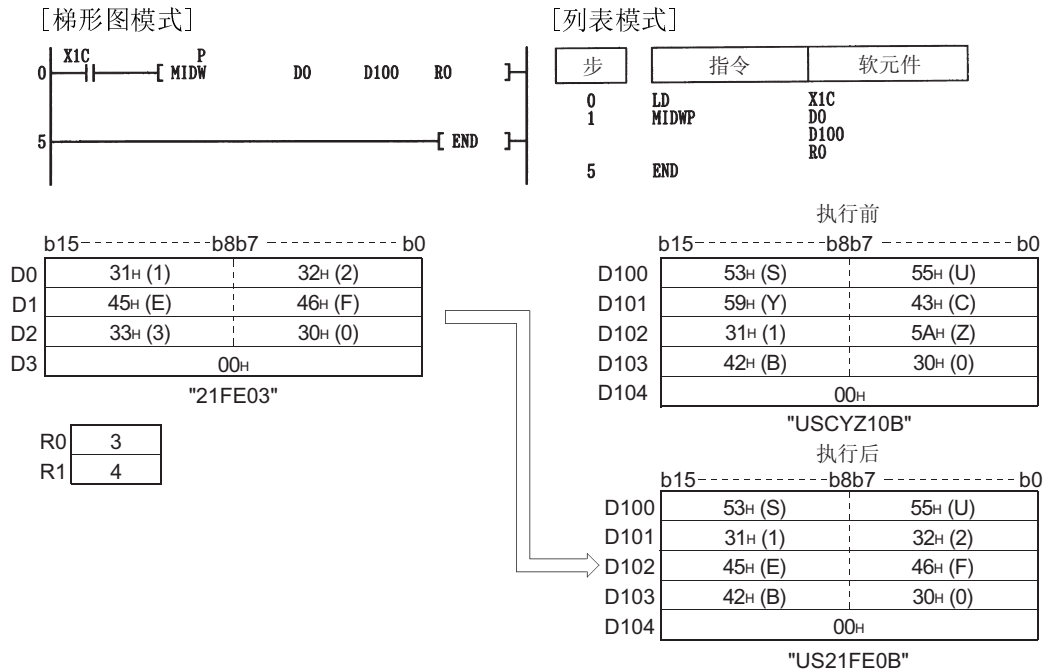
- ②的值超出了①的字符数。(错误代码: 4101)
- ②+1 的值超出了③的字符数。(错误代码: 4101)

[程序示例]

(1) 当 X0 为 ON 时，下列程序在从 D10 开始存储的字符串中，从左边起第 6 个字符处开始，将第 3 个字符存储到从 D0 开始的软元件中。



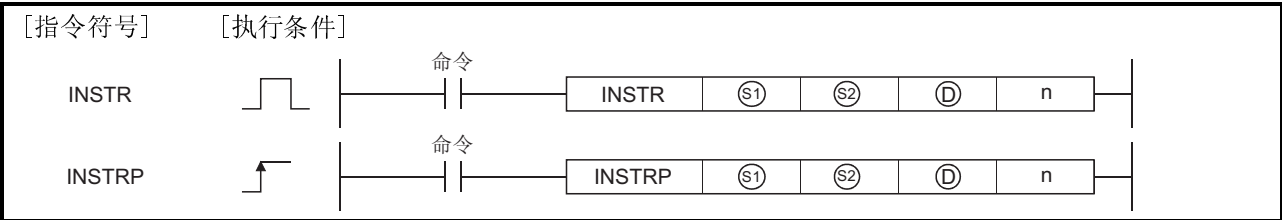
(2) 当 X1C 为 ON 时，下列程序将存储在从 D0 开始的软元件中的字符串数据的 4 个字符，存储到，从 D100 开始的软元件中的字符串数据，左起第 3 个字符开始的位置中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.11.17 字符串查找(INSTR, INSTRP)

设定 数据	可用软元件									
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[] []		特殊功能模 块 U[] [] G[] []	变址寄存器 Zn	常数		其它
	位	字		位	字			K, H	\$	
①	—	○		—				—	○	—
②	—	○		—				—	○	—
③	○	○		○				—	—	—
n	○	○		○				○	—	—



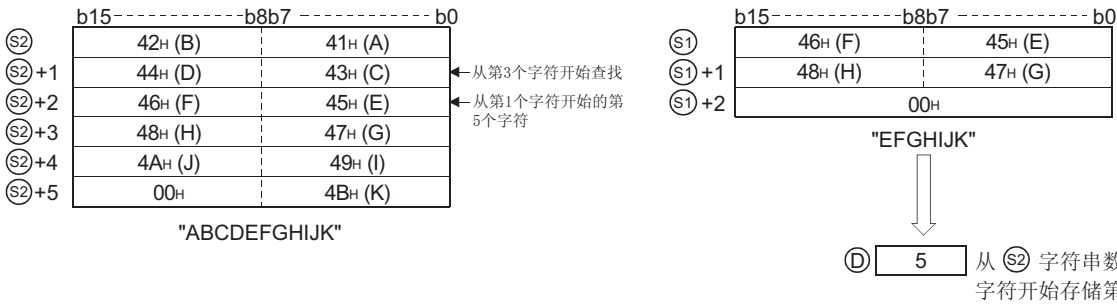
[设定数据]

设定数据	含义	数据类型
①	存储要查找的字符串的软元件的起始软元件号	字符串
②	存储被查找的字符串的软元件的起始软元件号	
③	存储查找结果的软元件的起始软元件号	BIN 16 位
n	开始查找的位置	

[功能]

- (1) 在由②指定号码的软元件开始的软元件中存储的字符串数据中，从左起第 n 个字符串数据开始，查找由①指定号码的软元件开始的软元件中存储的字符串数据，并且在由③指定的软元件中存储查找结果。
- 查找结果存储在从由③指定的字符串数据的起始字符开始的第 x 个字符中。

当 n=3



(2) 如果没有匹配的字符串数据，在⑤中存储“0”。

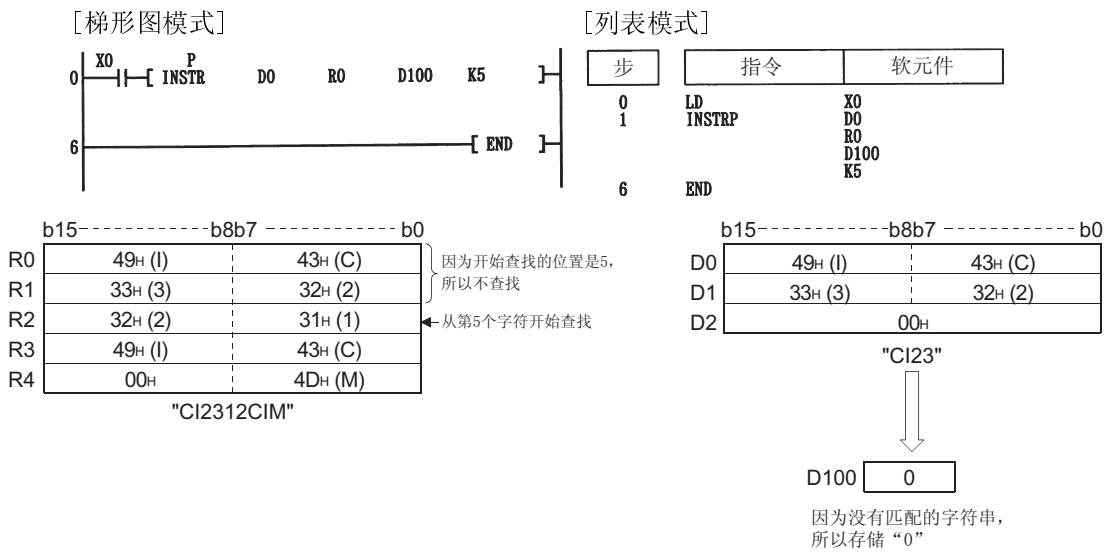
(3) 如果 n 是负值或“0”，将不执行任何处理。

[运行错误]

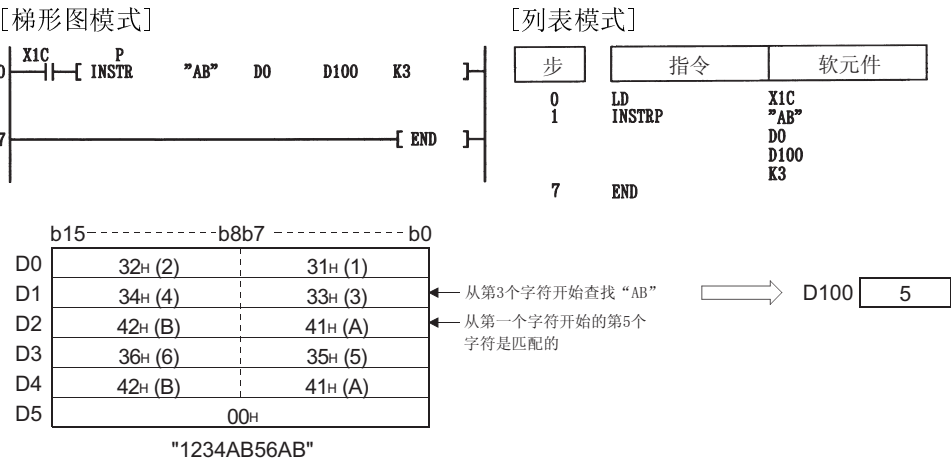
- (1) 在下列发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- n 的值超出了②的字符数 (错误代码: 4100)
 - 在由③、④指定的软元件之后的相应的软元件范围中不存在 00H (NULL)。 (错误代码: 4100)

[程序示例]

(1) 当 X0 为 ON 时，下列程序从 R0 开始的软元件中存储的字符串数据，左起第 5 个字符处开始查找从 D0 开始的软元件中的字符串数据，并且在 D100 中存储结果。



(2) 当 X1C 为 ON 时，下列程序从 D0 开始的软元件中存储的字符串数据，左起第 3 个字符处开始查找字符串数据“AB”，并且在 D100 中存储查找结果。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.11.18 浮点数到 BCD (EMOD, EMODP)

设定 数据	可用软元件									
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数		其它
	位	字		位	字			K, H	E	
①	—	○		—	○		—	—	○	—
②	○	○		○	○		○	○	—	—
③	—	○		—	—		—	—	—	—

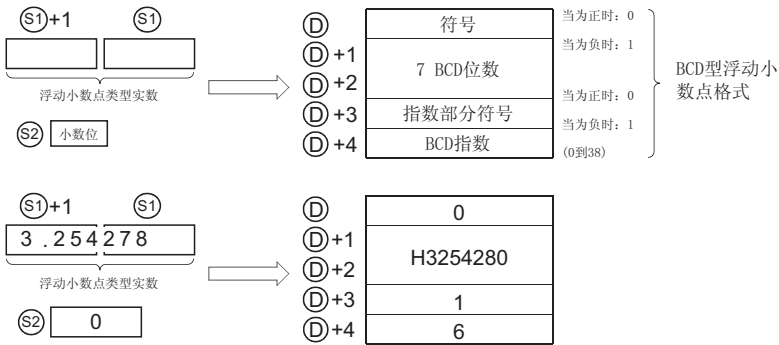
[指令符号]	[执行条件]
EMOD	
EMODP	

[设定数据]

设定数据	含义	数据类型
①	浮点型实数数据或存储浮点型实数数据的软元件的起始软元件号。	实数
②	小数部分位数或存储该类数据的软元件的起始软元件号。	BIN 16 位
③	将要存储 BCD 分析数据的软元件的起始软元件号	

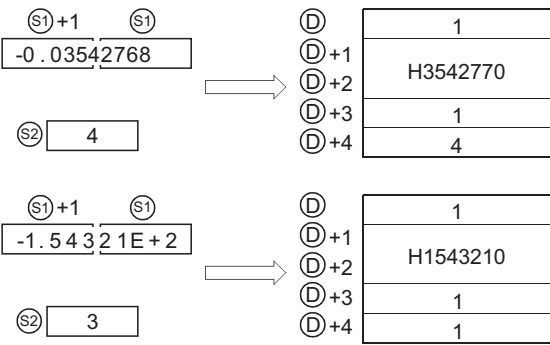
[功能]

- (1) 将在由①指定的软元件中存储的浮点类型实数，按照由②指定的软元件中存储的小数部分位数，分解为 BCD 浮点格式，并且从由③指定的软元件开始存储结果。

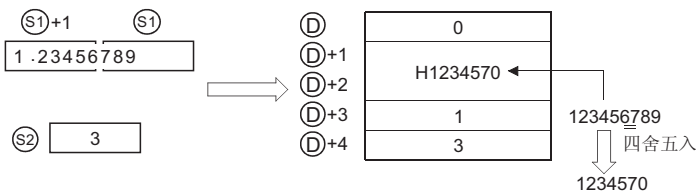


项目 ② 指定了项目 ① 的浮动小数点类型实数的小数部分的位数，在上述例子中，如下所示：

3.254278
 ^
 ^
 ^
② = 3



(2) 在D+1 和D+2 中存储的有效数字的第 7 位数被四舍五入，从而变为一个 6 位数。

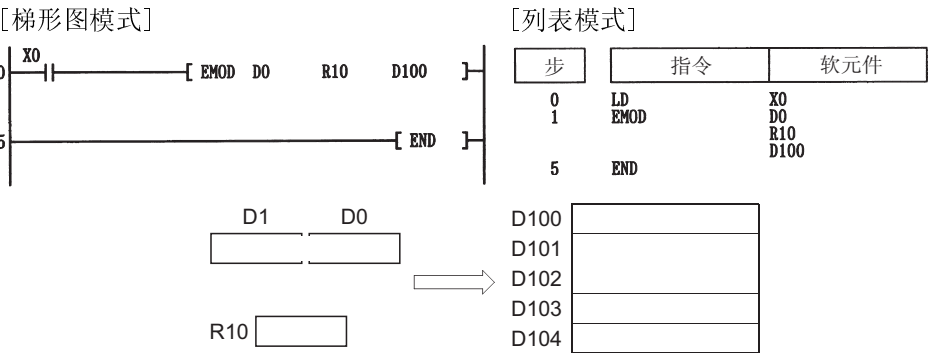


[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 由S2指定的小数部分位数在 0 到 7 的范围之外。(错误代码：4100)
 - 由D指定的软元件范围超出了相关软元件的范围(错误代码：4101)

[程序示例]



- (1) 当 X0 为 ON 时，下列程序将在 D0 和 D1 中存储的浮点类型实数数据，按照在 R10 中存储的值对应的小数部分的位数进行分解，并且将结果存储到从 D100 开始的软元件中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.11.19 从 BCD 格式数据到浮点数(EREXP, EREXPP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				U
①	—	○		—	—		—	—	—
②	○	○		○	○		○	○	—
③	—	○		—	○		—	—	—

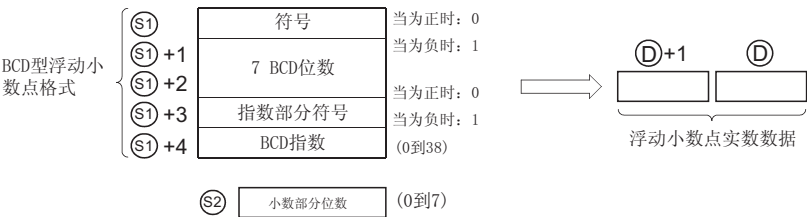
[指令符号]	[执行条件]
EREXP	
EREXPP	

[设定数据]

设定数据	含义	数据类型
①	存储 BCD 浮点格式数据的软元件的起始软元件号	BIN 16 位
②	小数部分位数或存储该类数据的软元件	BIN 16 位
③	将要存储浮点类型实数数据的软元件	实数

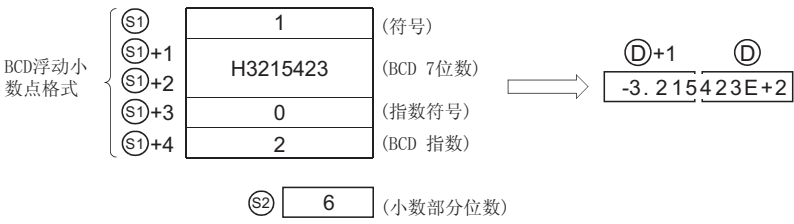
[功能]

(1) 将从软元件号由①指定的软元件开始的软元件中存储的 BCD 类型浮点格式数据，按照由②指定的软元件中存储的小数部分位数，转换成浮点类型实数，并且在从由③指定软元件号的软元件开始的软元件中存储结果。



- (2) 如果是正值，①的符号和①+3 的指数部分的符号设置为 0，如果是负值，设置为 1。
- (3) 对于①+4 个 BCD 指数，可以设置为 0 到 38 的值。

(4) 小数部分位数可以设置为 0 到 7。

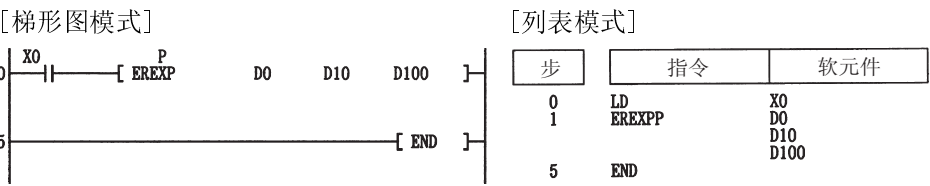


[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- (S1) 指定的格式指定既不是 0 也不是 1。 (错误代码：4100)
 - (S1)+1 和(S1)+2 指定的 BCD 数据超过了 8 位。 (错误代码：4100)
 - (S1)+3 指定的格式指定既不是 0 也不是 1。 (错误代码：4100)
 - (S1)+4 指定的指数数据在 0 到 38 的范围之外。 (错误代码：4100)
 - (S2)指定的小数部分位数在 0 到 7 的范围之外。 (错误代码：4100)

[程序示例]

- (1) 当 X0 为 ON 时，按照在 D10 中存储的小数部分位数，下列程序从将 D0 开始的软元件中存储的 BCD 浮点格式数据，转换为浮点类型实数数据，并且在 D100 和 D101 中存储结果。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

＊1：系列号的前五位数是04122或更大的数。

7.12 特殊功能指令

7.12.1 浮点数的 SIN 运算(SIN， SINP)

设定数据	可用软元件						
	内部软元件 (系统，用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○		—	○		—
⑥	—	○		—	○		—

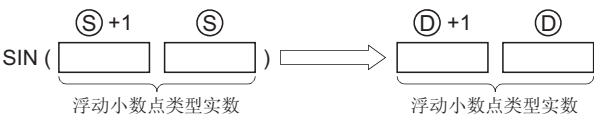
[指令符号]	[执行条件]
SIN	
SINP	

[设定数据]

设定数据	含义	数据类型
⑤	存储将执行 SIN(正弦) 运算的角度数据的软元件的起始软元件号	实数
⑥	存储运算结果的软元件的起始软元件号	

[功能]

(1) 计算由⑤指定的角度的 SIN（正弦）值，并且在⑥指定软元件号的软元件中存储运算结果。



(2) 由⑤指定的角是以弧度单位来设置的（度数×π/180）。
关于角度和弧度值之间的转换，请参见 RAD 和 DEG 指令。

[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 当指定的软元件包含-0 时*¹（错误代码：4100）

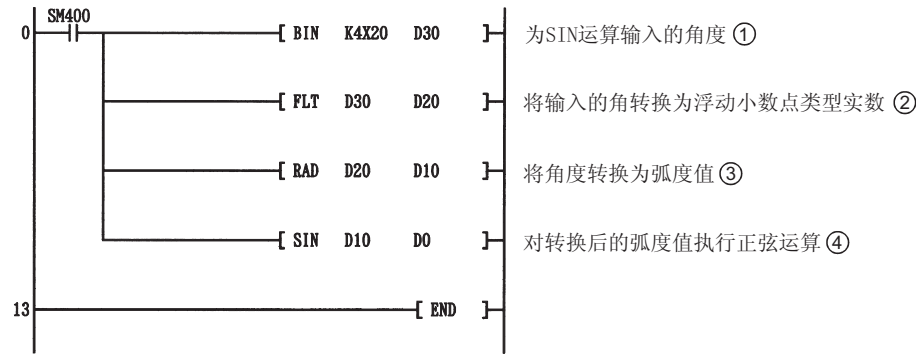
备注

*1：如果指定的是-0，也有不会导致运行错误的 CPU 模块。详细资料请参考章节 3.2.4。

[程序示例]

(1) 下列程序对存储在 X20 到 X2F 的 4 个 BCD 位数中的角度执行 SIN 运算。以浮点类型实数在 D0 和 D1 中存储结果。

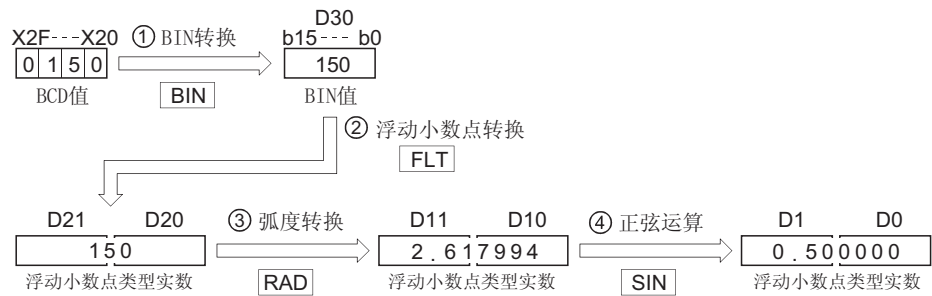
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	BIN	K4X20 D30
4	FLT	D30 D20
7	RAD	D20 D10
10	SIN	D10 D0
13	END	

[当 X20 到 X2F 指定的值为 150 时相关的运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

※1：系列号的前五位数是04122或更大的数。

7.12.2 浮点数的 COS 运算 (COS, COSP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 I[]\Q[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○		—	○		—
⑥	—	○		—	○		—

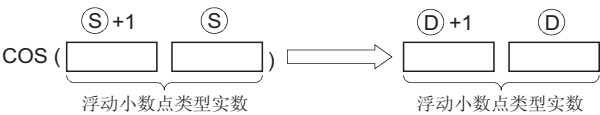
[指令符号]	[执行条件]
COS	
COSP	

[设定数据]

设定数据	含义	数据类型
⑤	存储执行 COS(余弦) 运算的角度数据的软元件的起始软元件号	实数
⑥	存储运算结果的软元件的起始软元件号	

[功能]

(1) 对由⑤指定的角度执行 COS (余弦) 运算, 并且在由⑥指定软元件号的软元件中存储运算结果。



(2) 由⑤指定的角度是以弧度单位来设置的(角度× $\pi/180$)。
关于角度和弧度值之间的转换, 请参见 RAD 和 DEG 指令。

[运行错误]

- (1) 在下列发生运行错误的情况中, 错误标志(SM0)变为 ON, 并且错误代码存储在 SD0 中。
- 当指定的软元件包含-0 时*1 (错误代码: 4100)

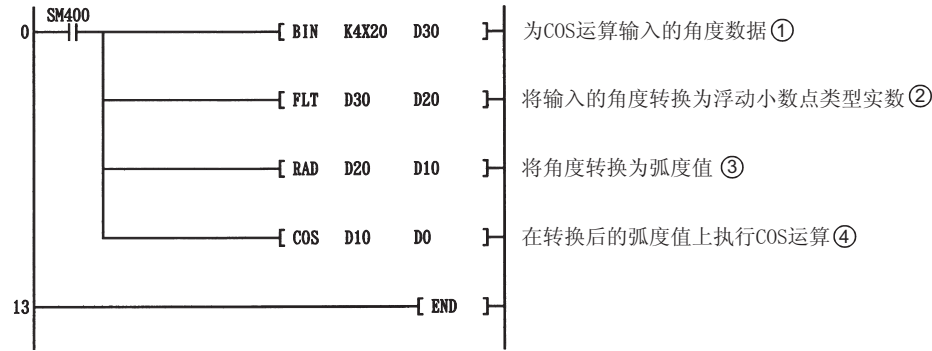
备注

*1: 如果指定的是-0, 也有不会导致运行错误的 CPU 模块。详细资料请参考章节 3.2.4。

[程序示例]

(1) 下列程序对存储在 X20 到 X2F 的 4 个 BCD 位数指定的角度数据执行 COS 运算，并且以浮点类型实数在 D0 和 D1 中存储结果。

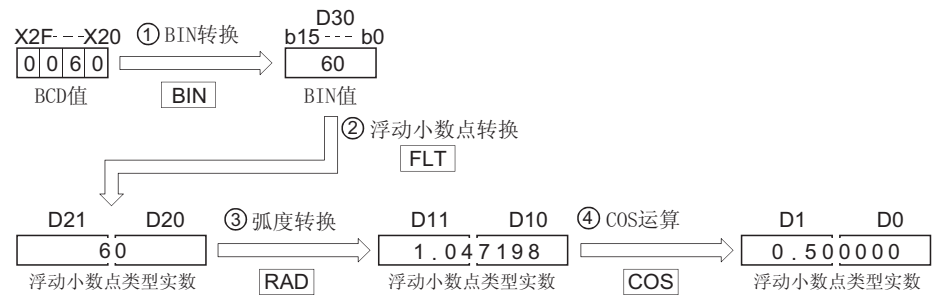
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	BIN	K4X20 D30
4	FLT	D30 D20
7	RAD	D20 D10
10	COS	D10 D0
13	END	

[当 X20 到 X2F 指定的值为 60 时相关的运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

＊1：系列号的前五位数是04122或更大的数。

7.12.3 浮点数的 TAN 运算 (TAN, TANP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○		—	○		—
⑥	—	○		—	○		—

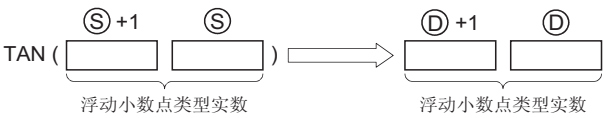
[指令符号]	[执行条件]
TAN	
TANP	

[设定数据]

设定数据	含义	数据类型
⑤	存储将要执行 TAN（正切）运算的角度数据的软元件的起始软元件号	实数
⑥	存储运算结果的软元件的起始软元件号	

[功能]

(1) 对由⑤指定的角度数据执行正切（TAN）运算，并且在由⑥指定的软元件中存储运算结果。



- (2) ⑤指定的角度是以弧度单位来设置的 (角度 $\times\pi/180$)。
关于角度和弧度值之间的转换，请参见 RAD 和 DEG 指令。
- (3) 当由⑤指定的角度是 $\pi/2$ 弧度，或 $(3/2)\pi$ 弧度时，在弧度值的计算当中将会产生一个运行错误，所以应该小心避免此类出错。

[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 运算结果在下列范围之外： (错误代码：4100)
 $0, \pm 2^{-126} \leq |\text{运算结果}| < \pm 2^{128}$
 - 当指定的软元件包含 - 0 时 *¹ (错误代码：4100)

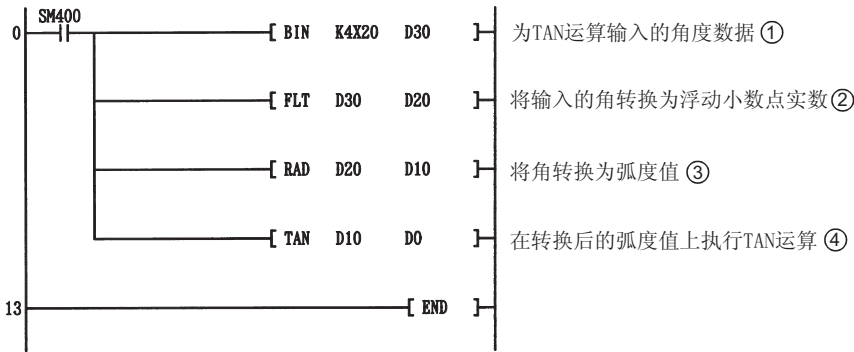
备注

*1: 如果指定的是-0，也有不会导致运行错误的 CPU 模块。详细资料请参考章节 3.2.4。

[程序示例]

- (1) 下列程序对由从 X20 到 X2F 的 4 个 BIN 位数设定的角度数据执行 TAN 运算，并且在 D0 和 D1 中存储结果为浮点类型实数。

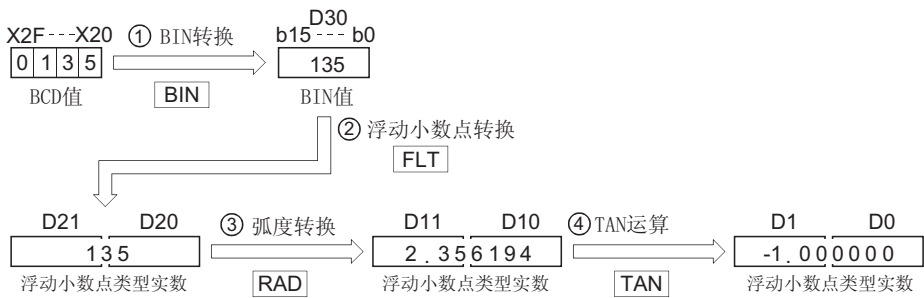
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	BIN	K4X20 D30
4	FLT	D30 D20
7	RAD	D20 D10
10	TAN	D10 D0
13	END	

[当 X20 到 X2F 的值指定为 135 时的相关运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.12.4 浮点数的 SIN⁻¹ 运算 (ASIN, ASINP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J□\□□		特殊功能模 块 U□\G□□	变址寄存器 Zn	常数 E	其它
	位	字		位	字				
⑤	—	○		—	○		—	○	—
⑥	—	○		—	○		—	—	—

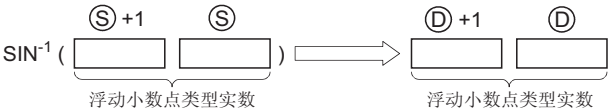
[指令符号]	[执行条件]
ASIN	
ASINP	

[设定数据]

设定数据	含义	数据类型
⑤	存储将要执行 SIN ⁻¹ (反正弦) 运算的 SIN 值的软元件的起始软元件号	实数
⑥	存储运算结果的软元件的起始软元件号	

[功能]

(1) 从由⑤指定的 SIN 值计算角度，并且在由⑥指定的字软元件中存储运算结果。



- (2) 由⑤指定的 SIN 值可在从-1.0 到 1.0 的范围之内。
- (3) 存储在⑥中的角度（运算结果）是以弧度单位来存储的。
关于角度和弧度值之间的转换，请参见 RAD 和 DEG 指令。

[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 由⑤指定的值在-1.0 到 1.0 的范围之外 (错误代码: 4100)
 - 当指定的软元件包含 - 0 时*¹ (错误代码: 4100)

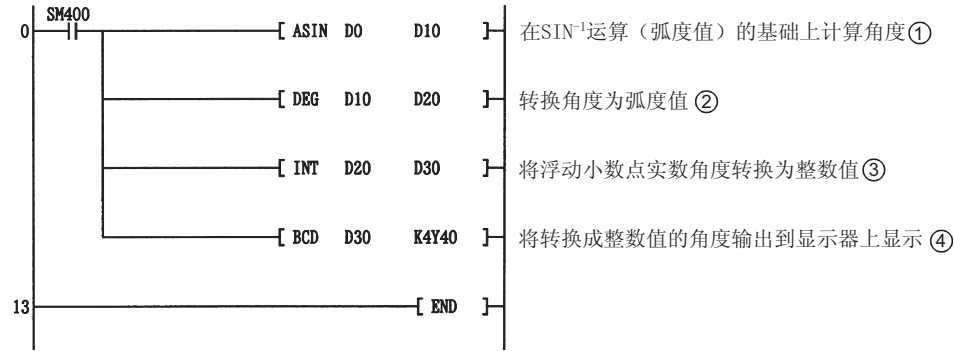
备注

*1: 如果指定的是-0，也有不会导致运行错误的 CPU 模块。详细资料请参考章节 3.2.4。

[程序示例]

(1) 下列程序查找 D0 和 D1 中的浮点类型实数的反正弦，并且输出角度到 Y40 到 Y4F 的 4 个 BCD 位上。

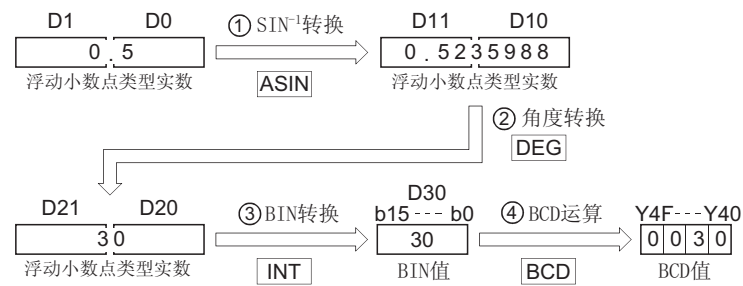
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	ASIN	D0 D10
4	DEG	D10 D20
7	INT	D20 D30
10	BCD	D30 K4Y40
13	END	

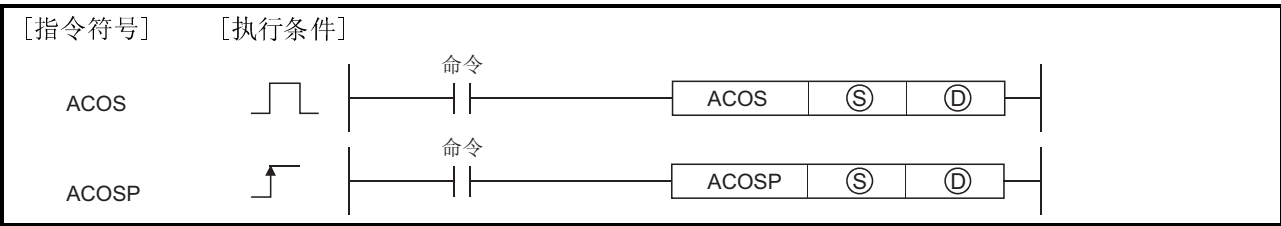
[当 D0 和 D1 的值是 0.5 时的相关运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.12.5 浮点数的 \cos^{-1} 运算 (ACOS, ACOSP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J□\□□		特殊功能模 块 U□\G□□	变址寄存器 Zn	常数 E	其它
	位	字		位	字				
⑤	—	○		—	○		—	○	—
⑥	—	○		—	○		—	—	—

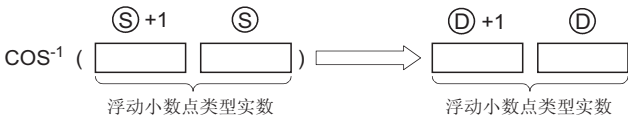


[设定数据]

设定数据	含义	数据类型
⑤	存储将要执行 \cos^{-1} (反余弦) 运算的 \cos 值的软元件的起始软元件号	实数
⑥	存储运算结果的软元件的起始软元件号	

[功能]

(1) 从由⑤指定的 \cos 值中计算角度，并且在由⑥指定的字软元件中存储运算结果。



(2) 由⑤指定的 \cos 值能在-1.0 到 1.0 的范围之内。

(3) 在⑥中存储的角度 (运算结果) 是按弧度单位来存储的。
关于角度数据和弧度之间转换的更多信息，请参见 DEG 和 RAD 指令的描述。

[运行错误]

- (1) 在下列发生运行错误的情况中，出错标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 由⑤指定的值在-1.0 到 1.0 的范围之外

(错误代码: 4100)

● 当指定的软元件包含 - 0 时*1

(错误代码: 4100)

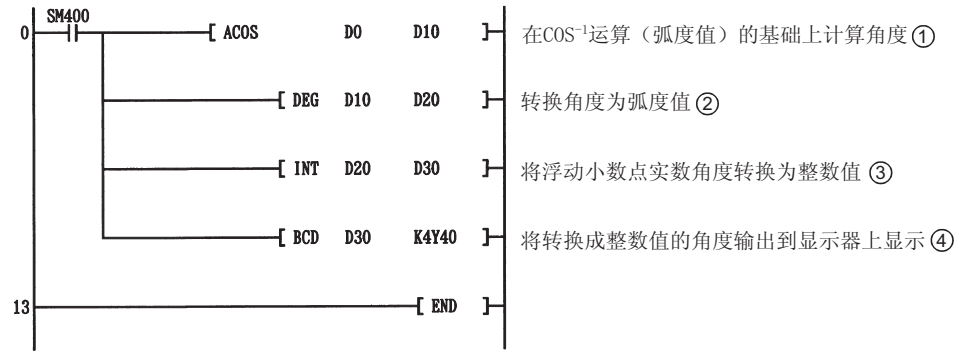
备注

*1: 如果指定的是-0，也有不会导致运行错误的 CPU 模块。详细资料请参考章节 3.2.4。

[程序示例]

(1) 下列程序查找 D0 和 D1 中的浮点类型实数的反余弦，并且输出角度到 Y40 到 Y4F 的 4 个 BCD 位上去。

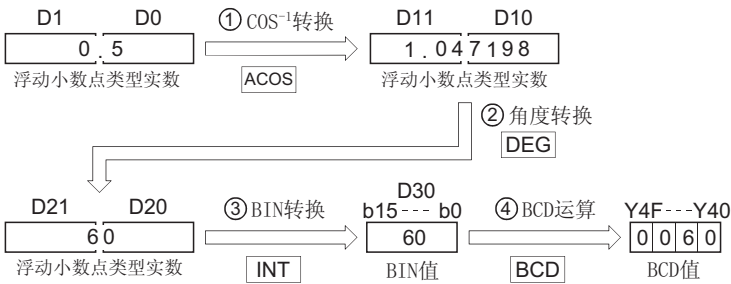
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	ACOS	D0 D10
4	DEG	D10 D20
7	INT	D20 D30
10	BCD	D30 K4Y40
13	END	

[当 D0 和 D1 的值是 0.5 时的相关运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.12.6 浮点数 TAN^{-1} 运算(ATAN, ATANP)

设定 数据	可用软元件								
	内部软元件 （系统，用户）		文件寄存器	MELSECNET/10(H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数 E	其它
	位	字		位	字				
⑤	—	○		—	○		—	○	—
⑥	—	○		—	○		—	—	—

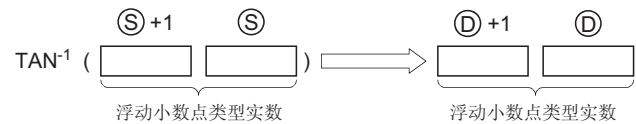
[指令符号]	[执行条件]
ATAN	
ATANP	

[设定数据]

设定数据	含义	数据类型
⑤	存储将要执行由 TAN^{-1} (反正切) 操作的 TAN 值的软元件的起始软元件号	实数
⑥	存储运算结果的软元件的起始软元件号	

[功能]

(1) 从由⑤指定的 TAN 值中计算角度，并且在由⑥指定的字软元件中存储运算结果。



(2) 在⑥中存储的角度（运算结果）是按弧度单位来存储的。
关于角度数据和弧度之间转换的更多信息，请参见 DEG 和 RAD 指令的描述。

[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SDO 中。
- 当指定的软元件包含 - 0 时*¹ (错误代码: 4100)

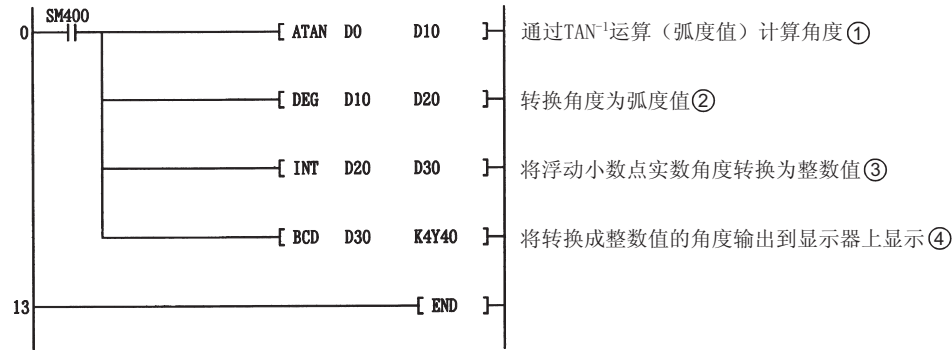
备注

*1: 如果指定的是-0，也有不会导致运行错误的 CPU 模块。详细资料请参考章节 3.2.4。

[程序示例]

(1) 下列程序查找 D0 和 D1 中的浮点类型实数的反正切，并且输出角度到 Y40 到 Y4F 的 4 个 BCD 位中去。

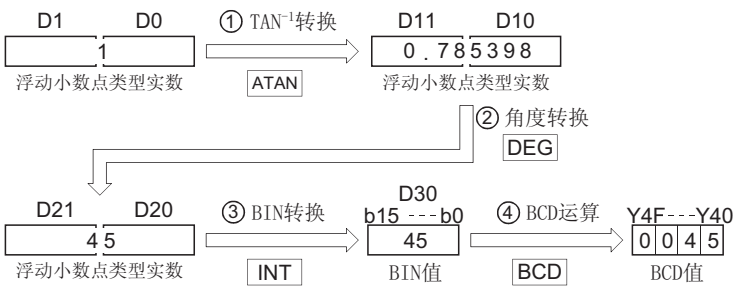
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	ATAN	D0 D10
4	DEG	D10 D20
7	INT	D20 D30
10	BCD	D30 K4Y40
13	END	

[当 D0 和 D1 的值是 1 时的相关运算]

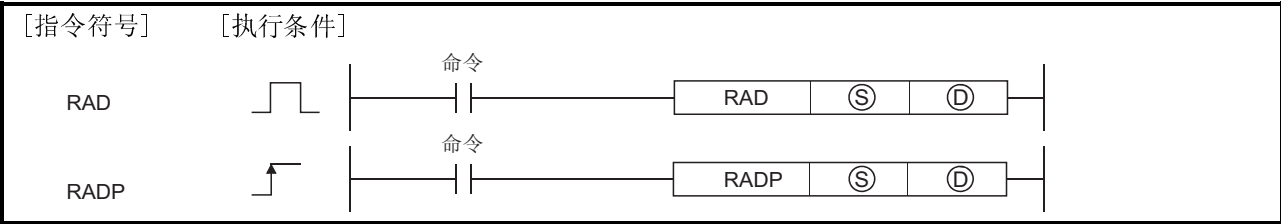


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

※1：系列号的前五位数是04122或更大的数。

7.12.7 从小数点角度到弧度的转换(RAD, RADP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J□\G□		特殊功能模 块 U□\G□	变址寄存器 Zn	常数 E	其它
	位	字		位	字				
⑤	—	○		—	○		—	○	—
⑥	—	○		—	○		—	—	—

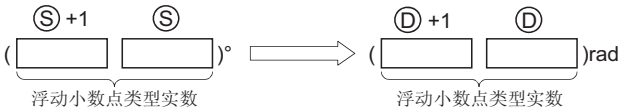


[设定数据]

设定数据	含义	数据类型
⑤	存储将要转换成弧度的角度的软元件的起始软元件号	实数
⑥	在转换结束之后存储弧度值的软元件的起始软元件号	

[功能]

- (1) 将角度大小的单位从由⑤指定的角度单位转换成弧度单位，并且在由⑥指定软元件号的软元件中存储结果。



- (2) 按照下列等式执行角度到弧度单位的转换。

弧度单位 = 角度单位 × $\frac{\pi}{180}$

[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 当指定的软元件包含 - 0 时※1 (错误代码：4100)

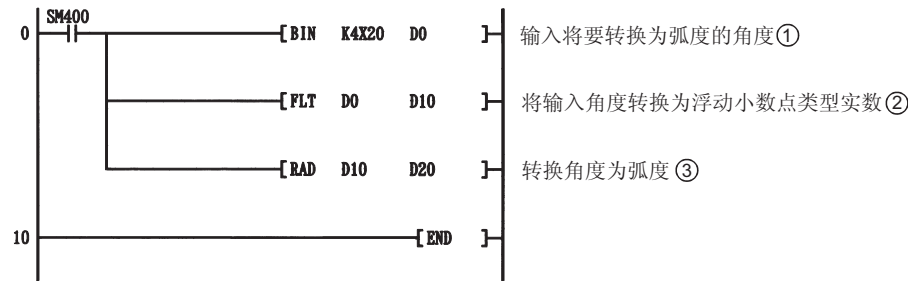
备注

※1：如果指定的是-0，也有不会导致运行错误的 CPU 模块。详细资料请参考章节 3.2.4。

[程序示例]

(1) 下列程序将在 X20 到 X2F 中的 4 个 BCD 位数设定的角度转换为弧度，并且在 D20 和 D21 中将结果存储为浮点类型实数。

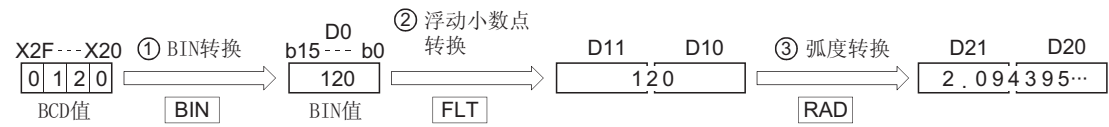
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	BIN	K4X20 D0
4	FLT	D0 D10
7	RAD	D10 D20
10	END	

[当 X20 到 X2F 指定的值为 120 时的相关运算]

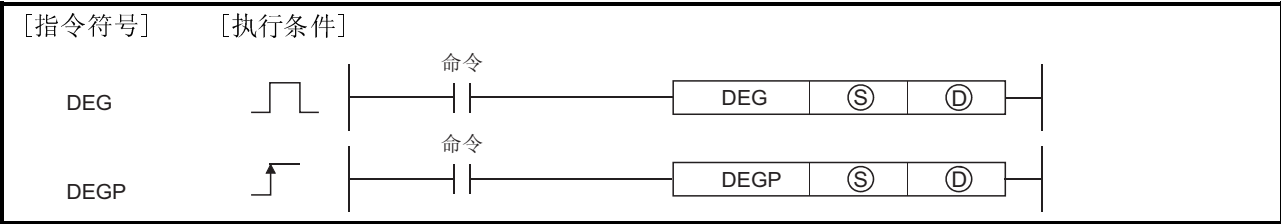


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

＊1：系列号的前五位数是04122或更大的数。

7.12.8 从浮点弧度到角度的转换(DEG, DEGP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数 E	其它
	位	字		位	字				
⑤	—	○		—	○		—	○	—
⑥	—	○		—	○		—	—	—

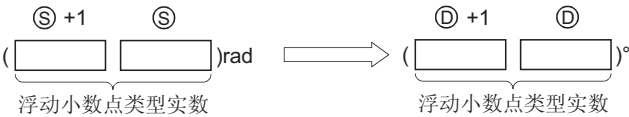


[设定数据]

设定数据	含义	数据类型
⑤	存储将要转换度数的弧度的软元件的起始软元件号	实数
⑥	在转换结束之后存储角度值的软元件的起始软元件号	

[功能]

- (1) 将角度大小的单位从由⑤指定的弧度单位转换为角度单位，并且在由⑥指定软元件号的软元件中存储结果。



- (2) 按照下列等式执行从弧度到角度的转换：

弧度单位 = 角度单位 × $\frac{180}{\pi}$

[运行错误]

- (1) 在下列发生运行错误的情况中，出错标志(SM0)变为 ON，并且错误代码存储在 SDO 中。
- 当指定的软元件包含 - 0 时＊1 (错误代码：4100)

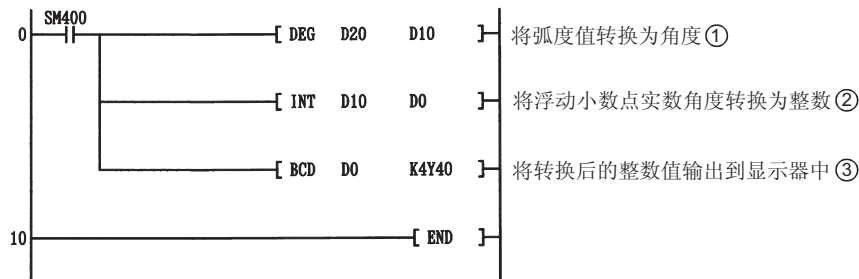
备注

＊1：如果指定的是-0，也有不会导致运行错误的 CPU 模块。详细资料请参考章节 3.2.4。

[程序示例]

(1) 下列程序将在 D20 和 D21 中的 BCD 位数上设置的浮点类型实数转换为角度，并且在 Y40 到 Y4F 中将结果存储为 BCD 值。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	DEG	D20 D10
4	INT	D10 D0
7	BCD	D0 K4Y40
10	END	

[当在 D20 和 D21 中指定的值为 1.435792 时的相关运算]

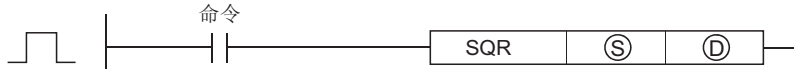
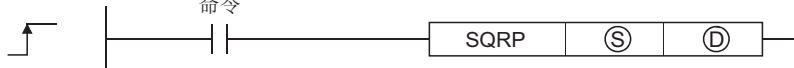


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

＊1：系列号的前五位数是04122或更大的数。

7. 12. 9 浮点数的平方根运算(SQR, SGRP)

设定 数据	可用的软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数 E	其它
	位	字		位	字				
⑤	—	○		—	○		—	○	—
⑥	—	○		—	○		—	—	—

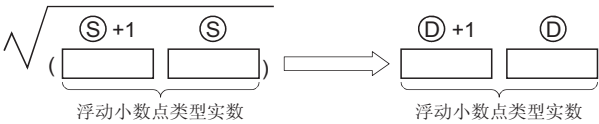
[指令符号]	[执行条件]
SQR	
SGRP	

[设定数据]

设定数据	含义	数据类型
⑤	存储将要执行平方根运算的数据的软元件的起始软元件号	实数
⑥	存储运算结果的软元件的起始软元件号	

[功能]

(1) 计算由⑤指定的值的平方根，并且在由⑥指定软元件号的软元件中存储运算结果。



(2) 只有正值才能由⑤指定。(在负值上不能执行运算)

[运行错误]

- (1) 在下列发生运行错误的情况中，出错标志(SM0)变为 ON，并且错误代码存储在 SDO 中。
- 由⑤指定的值是一个负值。
 - 当指定的软元件包含 - 0 时*1
- (错误代码：4100)

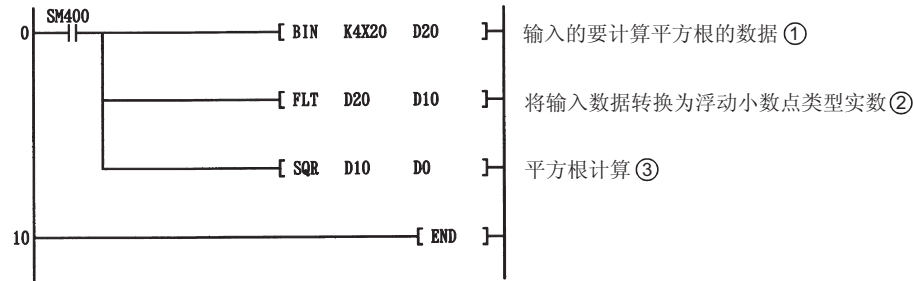
备注

*1: 如果指定的是-0，也有不会导致运行错误的 CPU 模块。详细资料请参考章节 3. 2. 4。

[程序示例]

(1) 下列程序查找由从 X20 到 X2F 的 4 个 BCD 位数设置的值的平方根，并且在 D0 和 D1 中将结果存储为浮点类型实数。

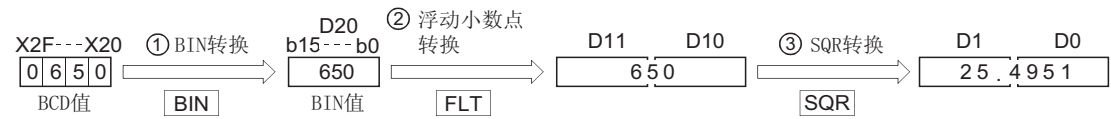
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	BIN	K4X20 D20
4	FLT	D20 D10
7	SQR	D10 D0
10	END	

[当由 X20 到 X2F 指定的值是 650 时的运算]

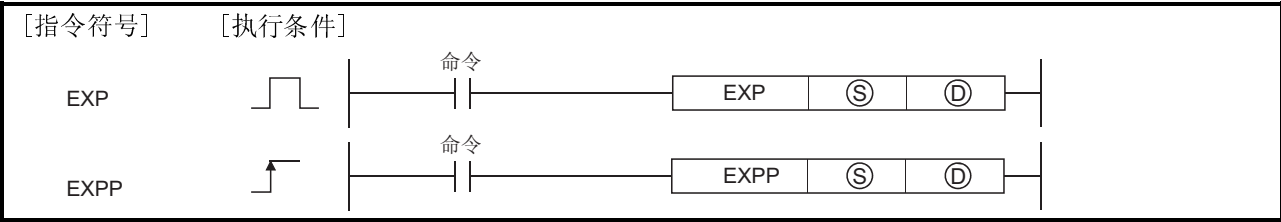


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

＊1：系列号的前五位数是04122或更大的数。

7.12.10 浮点数的指数运算(EXP, EXPP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○		—	○		—
⑥	—	○		—	○		—

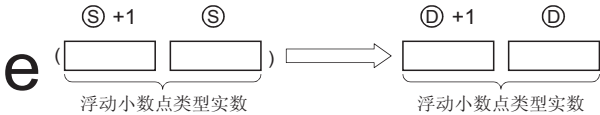


[设定数据]

设定数据	含义	数据类型
⑤	软元件的起始软元件号，在这些软元件中存储着指数运算将要执行的数据	实数
⑥	运算结果由存储的软元件的起始软元件号	

[功能]

(1) 计算由⑤指定值的指数，并且在由⑥指定的软元件中存储运算结果。



(2) 在指数运算中，底是按(e)是“2.71828”来计算的。

[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为ON，并且错误代码存储在SD0中。
- 运算结果不在下面表示的范围之内

(错误代码：4100)

$2^{-126} \leq |\text{运算结果}| < 2^{128}$
- 当指定的软元件包含-0时＊1

(错误代码：4100)

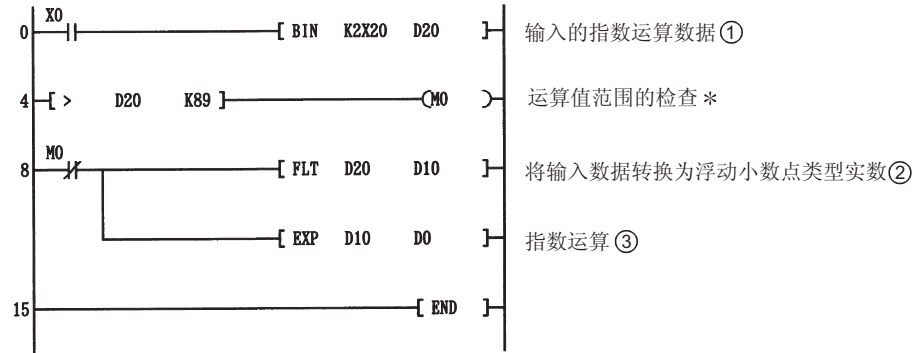
备注

＊1：如果指定的是-0，也有不会导致运行错误的CPU模块。详细资料请参考章节3.2.4。

[程序示例]

(1) 下列程序对由 X20 到 X27 中的 2 个 BCD 位数设置的值执行指数运算，并且在 D0 和 D1 中将结果存储为浮点类型实数。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	BIN	K2X20 D20
4	LD>	D20 K89
7	OUT	M0
8	LDI	M0
9	FLT	D20 D10
12	EXP	D10 D0
15	END	

[当由 X20 到 X27 指定的值为 13 时的相关运算]



要点
(1) *:如果 X20 到 X27 的 BCD 值小于 89，运算结果将在 2^{129} 以下，因为指数运算 $2^{129}=89.4$ 。 因为设置一个超过 90 的值将会返回一个运行错误，所以如果已经设置了一个超过 90 的值，则 M0 变为 ON 来避免出错。
(2) 将自然对数转换为常用对数。 在 CPU 模块中，使用自然对数来进行计算。 要获得一个常用对数值，将常用对数值⑤除以 0.43429。 $10^x = e^{\frac{x}{0.43429}}$

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

＊1：系列号的前五位数是04122或更大的数。

7.12.11 浮点数的自然对数运算(LOG, LOGP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J□\G□		特殊功能模 块 U□\G□	变址寄存器 Zn	常数 E	其它
	位	字		位	字				
⑤	—	○		—	○		—	○	—
⑥	—	○		—	○		—	—	—

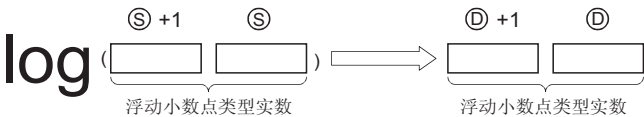
[指令符号]	[执行条件]
LOG	
LOGP	

[设定数据]

设定数据	含义	数据类型
⑤	存储用于自然对数运算的数据的软元件的起始软元件号	实数
⑥	存储运算结果的软元件的起始软元件号	

[功能]

(1) 以(e)作底计算由⑤指定的值的自然对数， 并且在由⑥指定的软元件中存储运算结果。



(2) 只有正值才能由⑤指定。(不能对负数执行运算。)

[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 由⑤指定的值是负值。(错误代码：4100)
 - 由⑤指定的值是 0。(错误代码：4100)
 - 当指定的软元件包含 - 0 时*1(错误代码：4100)

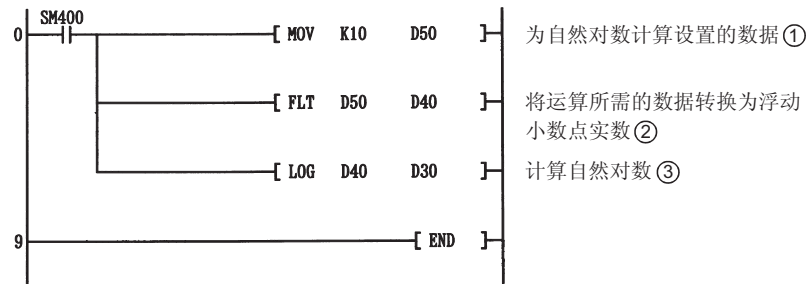
备注

*1: 如果指定的是-0，也有不会导致运行错误的 CPU 模块。详细资料请参考章节 3.2.4。

[程序示例]

(1) 下列程序查找由 D50 设置的值“10”的自然对数，并且在 D30 和 D31 中存储结果。

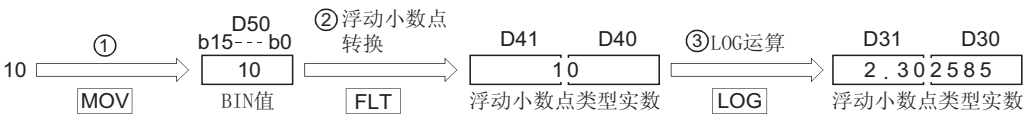
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	MOV	K10 D50
3	FLT	D50 D40
6	LOG	D40 D30
9	END	

[运算]



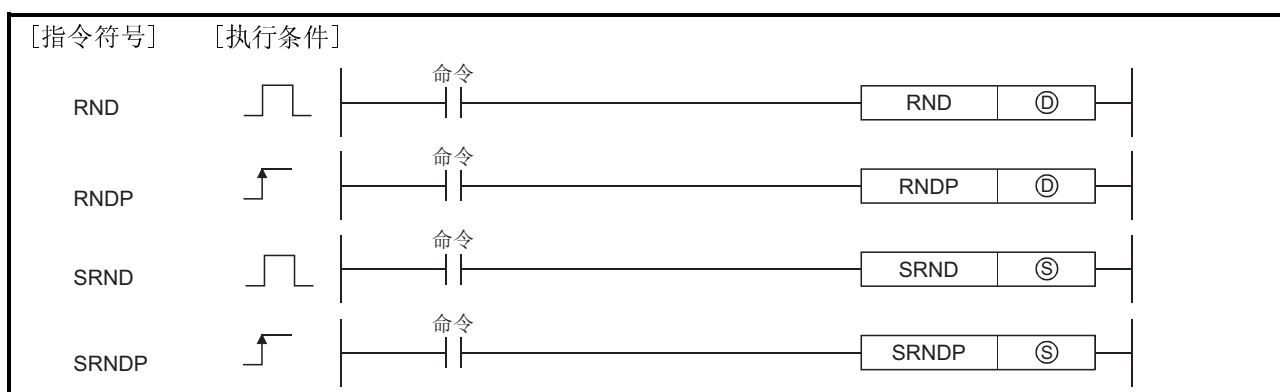
要点
(1) 从自然对数到常用对数的转换 CPU 模块在运算中使用自然对数。 如果要得到常用对数值，可以用下列表达式转换自然对数的值： $\log_{10}X = 0.43429 \times \log_e X$

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

*1 : 系列号的前五位数是04122或更大的数。

7.12.12 随机号的产生和系列更新(RND, RNDP, SRND, SRNDP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
④	○						—	—	
⑤	○						○	—	



[设定数据]

设定数据	含义	数据类型
④	存储随机号的软元件的起始软元件号	BIN 16 位
⑤	随机号系列数据或存储该类数据的软元件的起始软元件号	

[功能]

RND

从 0 到 32767 中产生随机号，并且存储在由Ⓓ指定的软元件中。

SRND

根据由⑤指定的软元件中存储的 16 位数据，更新随机号系列。

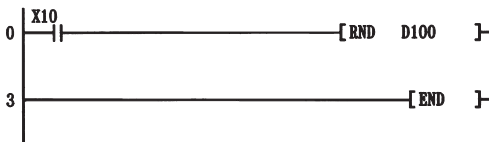
[运行错误]

没有与 RND (P) 或 SRND (P) 指令相关的运行错误。

[程序示例]

(1) 当 X10 为 ON 时，下列程序将随机号存储在 D100 中。

[梯形图模式]

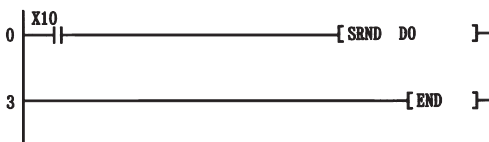


[列表模式]

步	指令	软元件
0	LD	X10
1	RND	D100
3	END	

(2) 当 X10 为 ON 时，下列程序根据 D0 的内容更新随机号系列。

[梯形图模式]



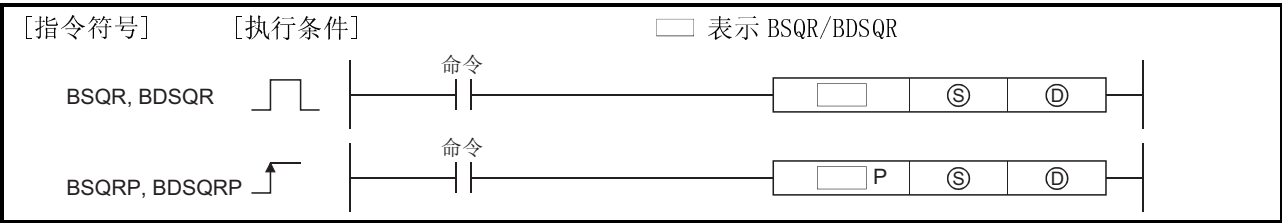
[列表模式]

步	指令	软元件
0	LD	X10
1	SRND	D0
3	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.12.13 BCD 4 位和 8 位平方根(BSQR, BSQRP, BDSQR, BDSQRP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J□□□		特殊功能模 块 U□□□□	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
⑤	○						○	—	
⑥	○						—	—	



[设定数据]

设定数据	含义	数据类型
⑤	将要执行平方根计算的数据或存储该类数据的软元件的软元件号	BCD 4-位/8-位
⑥	存储平方根计算结果的软元件的起始软元件号	BCD 4 位数

[功能]

BSQR

(1) 计算在⑤上指定的值的平方根，并且在由⑥指定软元件号的软元件中存储运算结果。

$\sqrt{\text{⑤}}$

=

⑥

⑥+1

整数部分

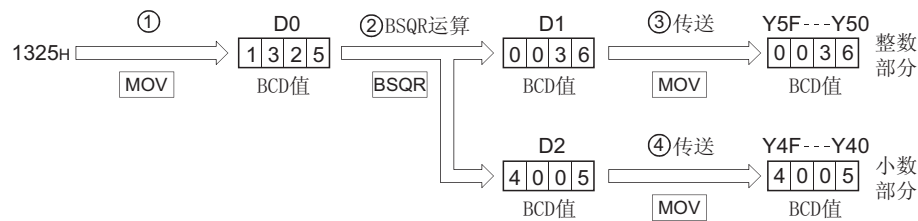
小数部分

(2) 能够在⑤中指定的值是最多 4 位数的 BCD 值（从 0 到 9999）。

(3) ⑥和⑥+1 的运算结果被存储为它们各自的 0 到 9999 之间的 BCD 值。

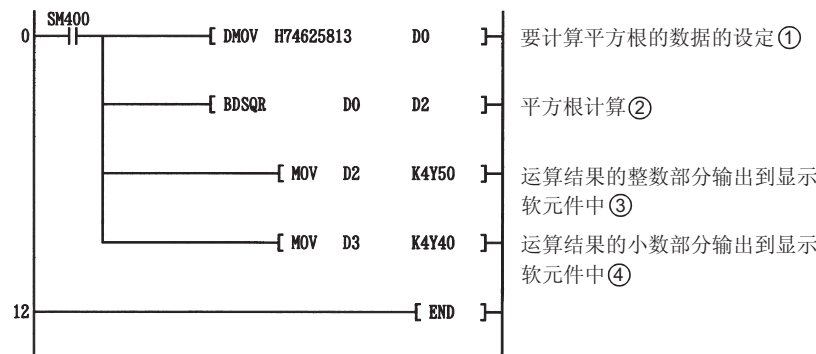
(4) 运算结果被统一为一个小数点后保留四位值。
因此，第四个小数位有±1 的误差。

[运算]



(2) 下列程序计算 BCD 值 74625813 的平方根，并输出结果的整数部分到从 Y50 到 Y5F 中的 4 个 BCD 位数上去，输出小数部分到从 Y40 到 Y4F 中的 4 个 BCD 位数上去。

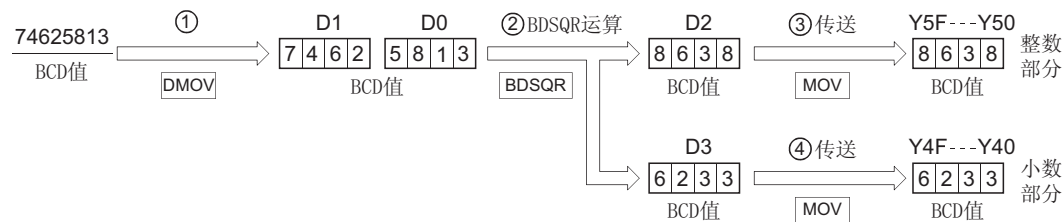
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	DMOV	H74625813
		D0
5	BDSQR	D0
		D2
8	MOV	D2
		K4Y50
10	MOV	D3
		K4Y40
12	END	

[运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.12.14 BCD 型 SIN 运算 (BSIN, BSINP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [J][N]		特殊功能模 块 [U][G]	变址寄存器 Zn
	位	字		位	字		
⑤	○	○				○	—
⑥	—	○				—	—

[指令符号]	[执行条件]
BSIN	
BSINP	

[设定数据]

设定数据	含义	数据类型
⑤	将要执行 SIN(正弦) 运算的数据, 或者存储该类数据的软元件的软元件号	BCD 4 位数
⑥	存储运算结果的软元件的起始软元件号	

[功能]

- (1) 计算由⑤指定的值(角度)的 SIN(正弦)值, 并在由⑥指定的软元件中存储运算结果的符号, 在由⑥+1 和⑥+2 指定的软元件中存储运算结果。
- SIN ⑤ =

⑥

⑥+1

⑥+2

符号

整数部分

小数部分
- (2) 由⑤指定的值是一个在 0 度和 360 度 (以度为单位) 之间的 BCD 值。
- (3) 如果结果是一个正值, 在⑥中存储的运算结果的符号将为 “0”, 如果结果是一个负值, 该符号为 “1”。
- (4) 存储在⑥+1 和⑥+2 的运算结果是在-1.000 和 1.000 之间的 BCD 值。
- (5) 运算结果在小数点后的第五位上四舍五入。

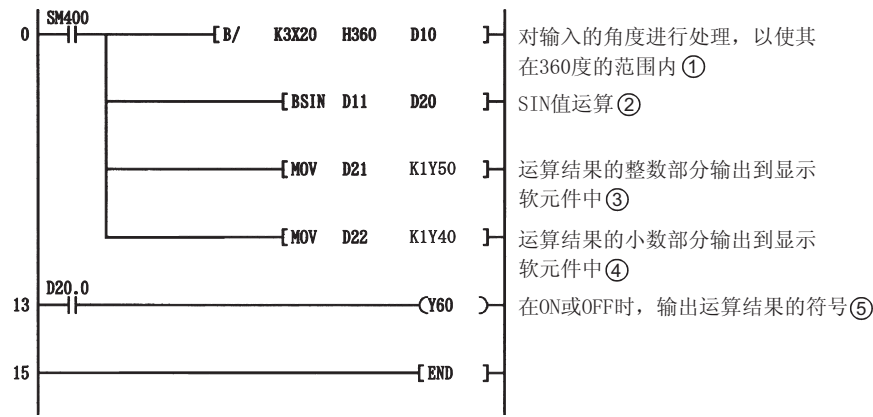
[运行错误]

- (1) 在下列发生运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 由⑤指定的数据不是一个 BCD 值。(错误代码: 4100)
 - 由⑤指定的数据不是在 0 到 360 的范围之内。(错误代码: 4100)

[程序示例]

- (1) 下面的程序示例计算由 X20 到 X2B 指定的 3 位 BCD 数据的 SIN 值，并且输出一个 1 位 BCD 的整数部分数据到 Y50 到 Y53，以及 4 位 BCD 数据的小数部分到 Y40 到 Y4F。
如果运算结果是负的，Y60 变为 ON。
(如果已经在 X20 到 X2F 中设置的值比 360 大，那么它将会被调整到 0 到 360 的范围里去。)

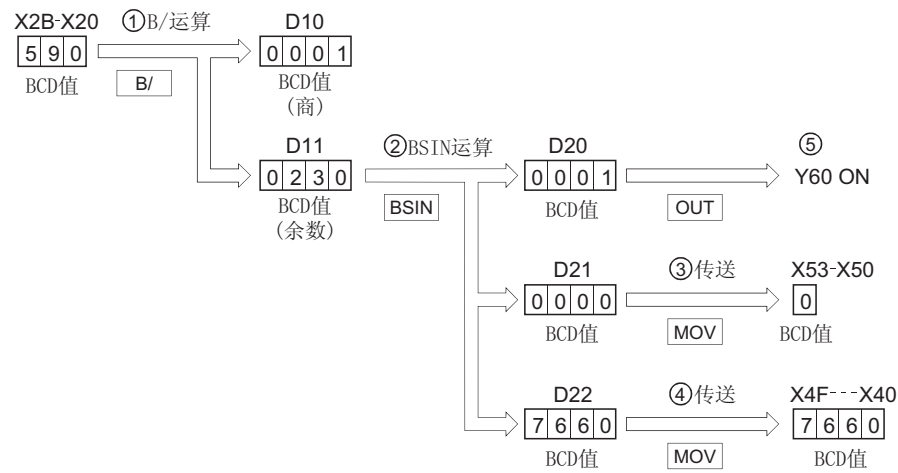
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	B/	K3X20 H360 D10
5	BSIN	D11 D20
8	MOV	D21 K1Y50
11	MOV	D22 K1Y40
13	LD	D20.0
14	OUT	Y60
15	END	

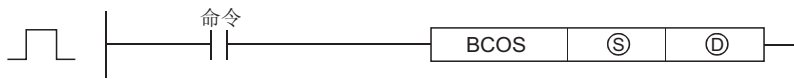
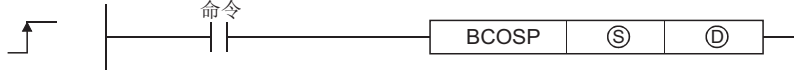
[由 X20 到 X2B 指定的值为 590 的相关运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.12.15 BCD 型 COS 运算(BCOS, BCOSP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [J][N][C]		特殊功能模 块 [U][V][G]	变址寄存器 Zn
	位	字		位	字		
⑤	○	○				○	—
⑥	—	○				—	—

[指令符号]	[执行条件]
BCOS	
BCOSP	

[设定数据]

设定数据	含义	数据类型
⑤	将要执行 COS (余弦) 运算的数据, 或存储该类数据的软元件的起始软元件号	BCD 4 位
⑥	存储运算结果的软元件的起始软元件号	

[功能]

(1) 计算由⑤指定的值 (角度) 的 COS (余弦) 值, 然后在由⑥指定的字软元件中存储运算结果的符号, 在由⑥+1 和⑥+2 指定的字软元件中存储运算结果。

$$\text{COS } \textcircled{5} =$$

⑥	⑥+1	⑥+2
符号	整数部分	小数部分

- (2) 由⑤指定的值是一个在 0 度到 360 度之间的 BCD 值 (单位为度)。
- (3) 如果结果是正值, 存储在⑥中的运算结果的符号将为 “0”, 如果是负值, 这个符号将为 “1”。
- (4) 存储在⑥+1 和⑥+2 中的运算结果为在-1.000 和 1.000 之间的 BCD 值。
- (5) 运算结果在小数点后第五位上四舍五入。

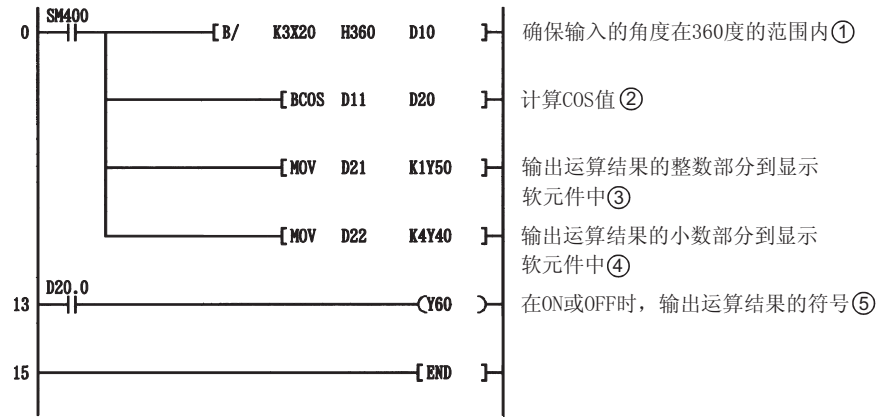
[运行错误]

- (1) 在下列发生运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 由⑤指定的数据不是一个 BCD 值。 (错误代码: 4100)
 - 由⑤指定的数据不在 0 到 360 的范围之内。 (错误代码: 4100)

[程序示例]

(1) 下列程序计算从 X20 到 X2B 的 3 个 BCD 位指定的数据的余弦，并且输出结果的 1 个 BCD 位的整数部分到 Y50 到 Y53，输出结果的 4 个 BCD 位的小数部分到从 Y40 到 Y4F。
如果运算结果为负，Y60 变为 ON。

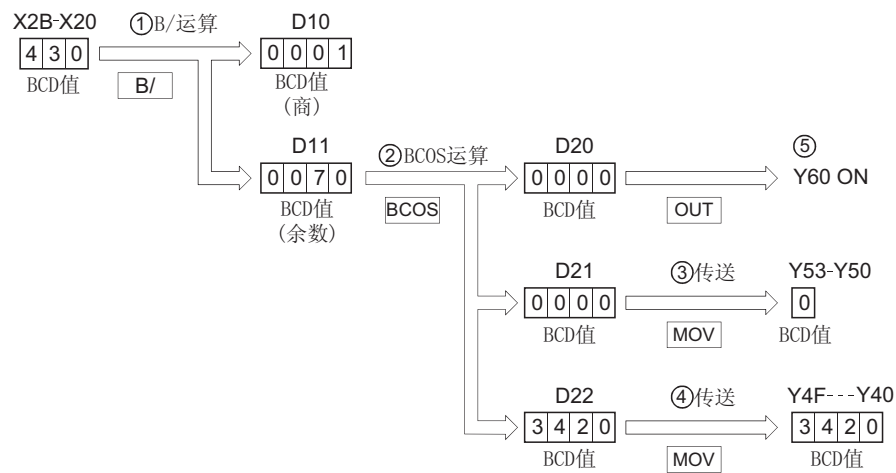
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	SM400
1	B/	K3X20 H360 D10
5	BCOS	D11 D20
8	MOV	D21 K1Y50
11	MOV	D22 K4Y40
13	LD	D20.0
14	OUT	Y60
15	END	

[当由 X20 到 X2B 指定的值为 430 时的相关运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.12.16 BCD 型 TAN 运算(BTAN, BTANP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	○	○				○	—
⑥	—	○				—	—

[指令符号]	[执行条件]
BTAN	
BTANP	

[设定数据]

设定数据	含义	数据类型
⑤	将要执行 TAN(正切) 运算的数据, 或者存储该类数据的软元件的起始软元件号	BCD 4 位数
⑥	存储运算结果的软元件的起始软元件号	

[功能]

- (1) 计算由⑤指定的值(角度)的 TAN(正切) 值, 并且在由⑥指定的字软元件中存储运算结果的符号, 且在由⑥+1 和⑥+2 指定的字软元件中存储运算结果。
- TAN⑤ =

⑥
符号

⑥+1
整数部分

⑥+2
小数部分
- (2) 由⑤指定的值是一个在 0 度和 360 度之间的 BCD 值(单位为度)。
- (3) 如果结果是一个正值, 在⑥中存储的运算结果的符号将为“0”, 如果结果是一个负值, 该符号为“1”。
- (4) 存储在⑥+1 和⑥+2 中的运算结果是在从-57.2900 到 57.2900 范围内的 BCD 值。
- (5) 运算结果在小数点后第五位上四舍五入。

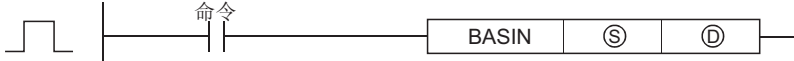
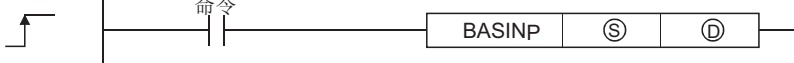
[运行错误]

- (1) 在下列发生运行错误的情况中, 错误标志(SM0)变为 ON, 并且错误代码存储在 SD0 中。
- 由⑤指定的数据不是一个 BCD 值。(错误代码: 4100)
 - 由⑤指定的数据不在 0 到 360 的范围之内。(错误代码: 4100)
 - 由⑤指定的数据为 90°或 270°。(错误代码: 4100)

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7. 12. 17 BCD 型 SIN⁻¹ 运算 (BASIN, BASINP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
⑤	—	○		—				—	—
⑩	○	○		○				—	—

[指令符号]	[执行条件]
BASIN	
BASINP	

[设定数据]

设定数据	含义	数据类型
Ⓔ	将要执行 SIN ⁻¹ (反正弦) 运算的数据, 或存储该类数据的软元件号	BCD 4 位数
Ⓕ	存储运算结果的软元件的起始软元件号	

[功能]

(1) 对由Ⓔ指定的值执行 SIN⁻¹ (反正弦) 运算, 并且在Ⓕ指定的软元件中存储运算结果 (角度)。

$$\text{SIN}^{-1} = (\begin{array}{|c|} \hline \text{Ⓔ} \\ \hline \text{符号} \\ \hline \end{array} \begin{array}{|c|} \hline \text{Ⓔ}+1 \\ \hline \text{整数部分} \\ \hline \end{array} . \begin{array}{|c|} \hline \text{Ⓔ}+2 \\ \hline \text{小数部分} \\ \hline \end{array}) = \text{Ⓕ}$$

(2) 在Ⓔ中设置运算数据的符号。
如果运算数据是正值, 符号设置为 “0”, 如果为负值, 符号设置为 “1”。

(3) 在小数点和小数部分前面的部分分别以 BCD 值存储在Ⓔ+1 和Ⓔ+2 中 (可以在 0 和 1.0000. 中设置)。

(4) 存储在Ⓕ中的运算结果为 0 度到 90 度和 270 度到 360 度之间的 BCD 值 (单位为度)。

(5) 计算结果是一个小数部分已经被四舍五入的值。

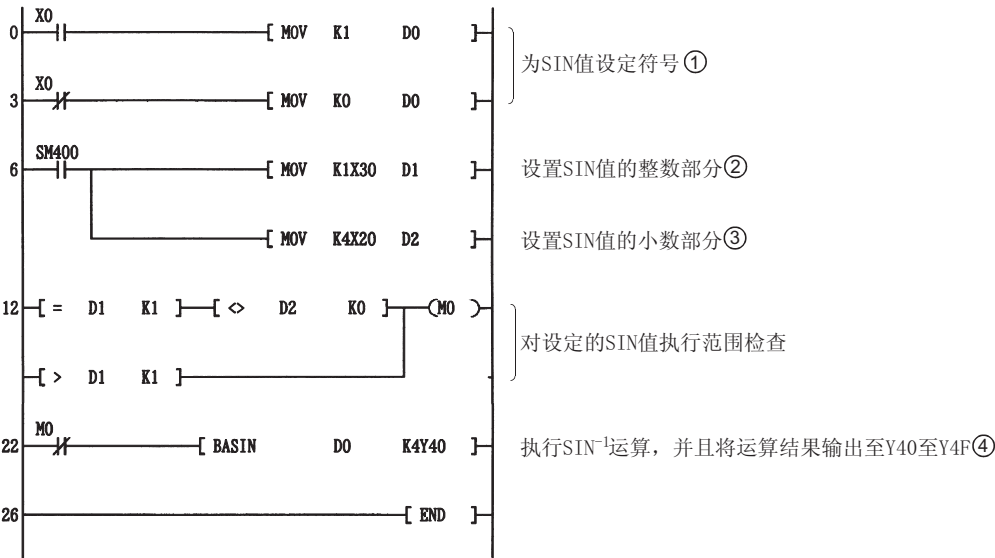
[运行错误]

- (1) 在下列发生运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 由Ⓔ指定的数据不是一个 BCD 值。 (错误代码: 4100)
 - 由Ⓔ指定的数据不在从-1.000 到 1.000 的范围之内 (错误代码: 4100)

[程序示例]

(1) 下列程序对符号(当 X0 是 OFF 时为正, 当 X0 是 ON 时为负), 从 X30 到 X33 的 1 位 BCD 整数部分和从 X20 到 X2F 的 4 位 BCD 小数部分执行 SIN^{-1} 运算, 并且输出从 Y40 到 Y4F 的 4 个 BCD 位中的计算出的角度。

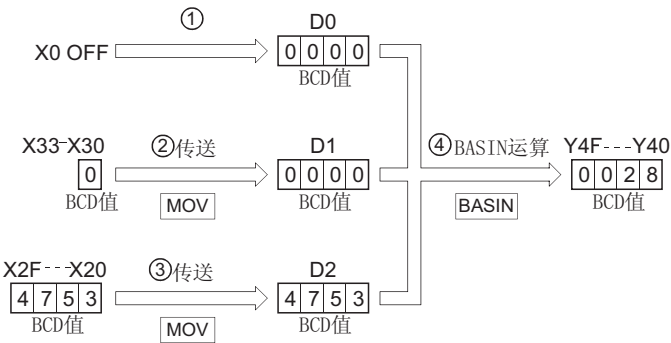
[梯形图模式]



[指令模式]

步	指令	软元件
0	LD	X0
1	MOV	K1 D0
3	LDI	X0
4	MOV	K0 D0
6	LD	SM400
7	MOV	K1X30 D1
10	MOV	K4X20 D2
12	LD=	D1 K1
15	AND<>	D2 K0
18	OR>	D1 K1
21	OUT	M0
22	LDI	M0
23	BASIN	D0 K4Y40
26	END	

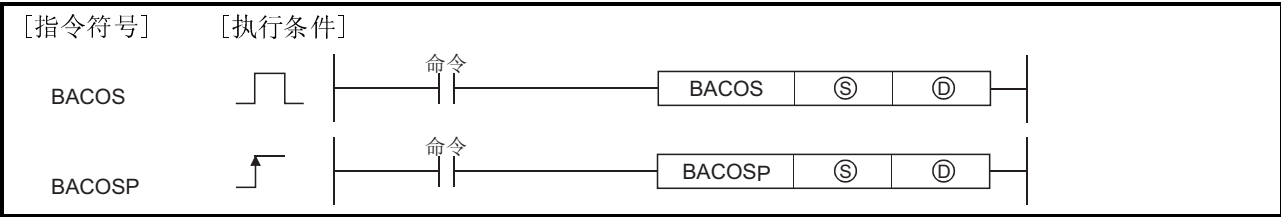
[当 X20 到 X33 指定的值为 0.4753 时的相关运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7. 12. 18 BCD 型 COS^{-1} 运算 (BACOS, BACOSP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\C[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数	其它
	位	字		位	字				
⑤	—	○		—				—	—
⑩	○	○		○				—	—



[设定数据]

设定数据	含义	数据类型
Ⓔ	将要执行 COS^{-1} (反余弦) 的数据, 或存储该类数据的软元件的起始软元件号	BCD 4 位数
Ⓕ	存储运算结果的软元件的起始软元件号	

[功能]

(1) 在由Ⓔ指定的值上执行 COS^{-1} (反余弦) 运算, 并且在Ⓕ指定的软元件中存储运算结果。

Ⓔ

Ⓔ+1

Ⓔ+2

$$\text{COS}^{-1} = (\begin{array}{|c|c|c|} \hline \text{符号} & \text{整数部分} & \text{小数部分} \\ \hline \end{array}) = \text{Ⓕ}$$

- (2) 在Ⓔ中设置运算数据的符号。
如果运算数据是正值, 符号设置为 “0”, 如果为负值, 符号设置为 “1”。
- (3) 整数部分和小数部分前面的部分被分别以 BCD 值存储在Ⓔ+1 和Ⓔ+2 中。
(可以在 0 和 1.0000. 中设置)
- (4) 存储在Ⓕ中的运算结果是一个在 0 度到 180 度 (单位为度) 之间的 BCD 值。
- (5) 计算结果是一个小数已经被四舍五入的值。

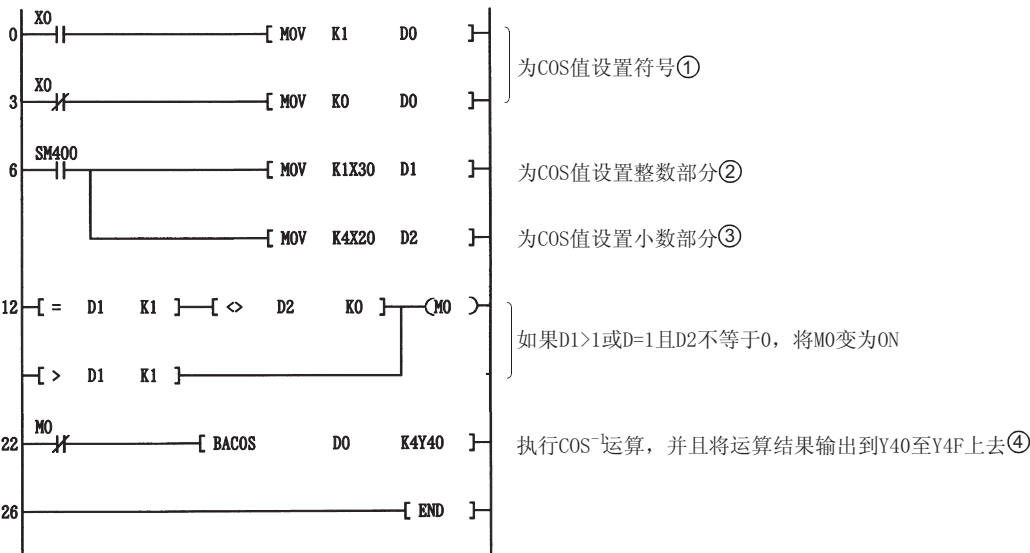
[运行错误]

- (1) 在下列发生运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 由Ⓔ指定的运算数据不是一个 BCD 值。 (错误代码: 4100)
 - 由Ⓔ指定的运算数据不在从 -1.000 到 1.000 的范围之内。 (错误代码: 4100)

[程序示例]

(1) 下列程序对符号(当 X0 是 OFF 时为正, 当 X0 是 ON 时为负), 从 X30 到 X33 的 1 位 BCD 整数部分和从 X20 到 X2F 的 4 位 BCD 小数部分执行 COS^{-1} 运算, 并且输出从 Y40 到 Y4F 的 4 个 BCD 位中的计算出的角度。

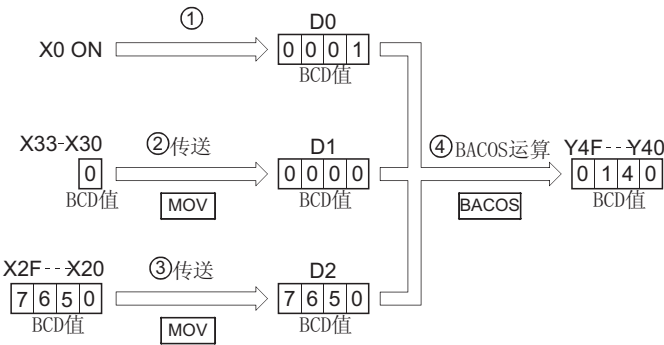
[梯形图模式]



[指令模式]

步	指令	软元件
0	LD	X0
1	MOV	K1
		D0
3	LDI	X0
4	MOV	K0
		D0
6	LD	SM400
7	MOV	K1X30
		D1
10	MOV	K4X20
		D2
12	LD=	D1
		K1
15	AND<>	D2
		K0
18	OR>	D1
		K1
21	OUT	M0
22	LDI	M0
23	BACOS	D0
		K4Y40
26	END	

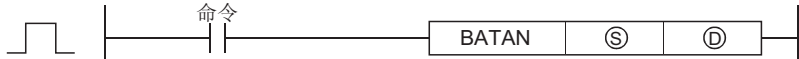
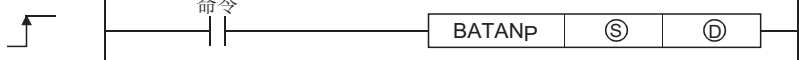
[如果 X0 和 X20 到 X33 指定的值为-0.7650 时的相关运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7. 12. 19 BCD 型 TAN⁻¹ 运算 (BATAN, BATANP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○		—			—
⑥	○	○		○			—

[指令符号]	[执行条件]
BATAN	
BATANP	

[设定数据]

设定数据	含义	数据类型
⑤	将要执行 TAN ⁻¹ (反正切) 运算的数据, 或存储该类数据的软元件的起始软元件号。	BCD 4 位数
⑥	存储运算结果的软元件的起始软元件号。	

[功能]

(1) 在由⑤指定的值上执行 TAN⁻¹ (反正切) 运算, 并且在⑥指定的软元件中存储运算结果 (角度)。

$$\text{TAN}^{-1} = (\begin{matrix} \text{⑤} \\ \text{符号} \end{matrix} \begin{matrix} \text{⑤+1} \\ \text{整数部分} \end{matrix} \begin{matrix} \text{⑤+2} \\ \text{小数部分} \end{matrix}) = \text{⑥}$$

(2) 在⑤中设置运算数据的符号。
如果运算数据是正值, 符号设置为 “0”, 如果为负值, 符号设置为 “1”。

(3) 在小数点和小数部分前面的部分被分别以 BCD 值存储在⑤+1 和⑤+2 中。
(可以在从 0 到 9999.9999 的范围中设置此数值。)

(4) 存储在⑥中的运算结果为 0 度到 90 度和 270 度到 360 度之间的 BCD 值 (单位为度)。

(5) 计算结果是一个小数已经被四舍五入的值。

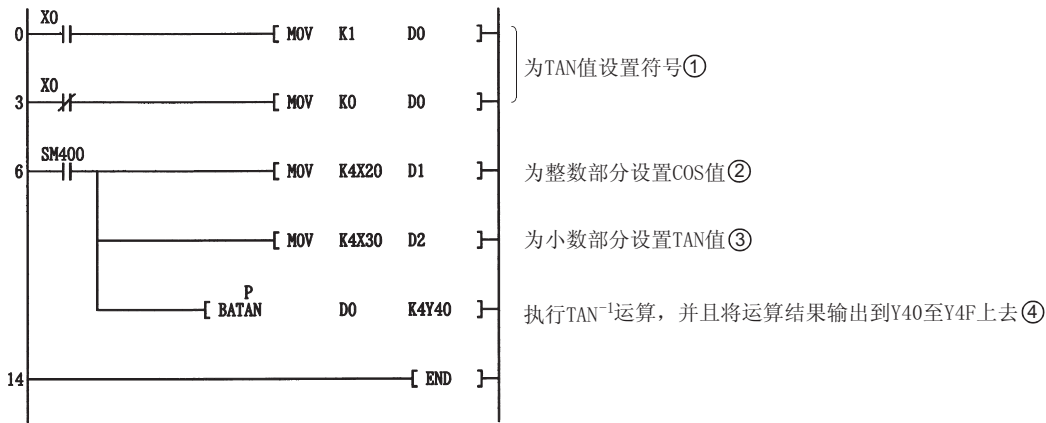
[运行错误]

- (1) 在下列发生运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 由⑤指定的运算数据不是一个 BCD 值。 (错误代码: 4100)

[程序示例]

(1) 下列程序对符号(当 X0 是 OFF 时为正, 当 X0 是 ON 时为负), 从 X20 到 X2F 的 4 位 BCD 整数部分和从 X30 到 X3F 的 4 位 BCD 小数部分执行 TAN^{-1} 运算, 并且输出从 Y40 到 Y4F 的 4 个 BCD 位中的计算出的角度。

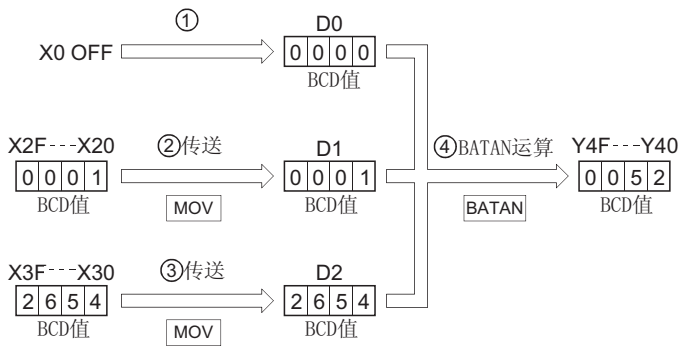
[梯形图模式]



[指令模式]

步	指令	软元件
0	LD	X0
1	MOV	K1
		D0
3	LDI	X0
4	MOV	K0
		D0
6	LD	SM400
7	MOV	K4X20
		D1
9	MOV	K4X30
		D2
11	BATANP	D0
		K4Y40
14	END	

[当 X0 和 X20 到 X2F 的值指定为 1.2654 时的相关运算]

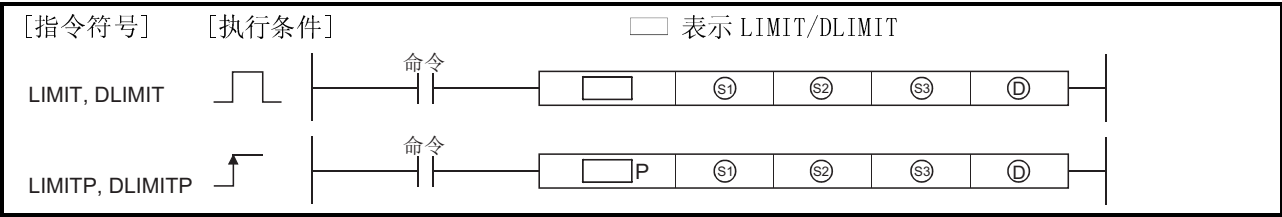


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.13 数据控制指令

7.13.1 BIN 16 位和BIN 32 位数据的高低限控制
(LIMIT, LIMITP, DLIMIT, DLIMITP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
①	○						○	—	
②	○						○	—	
③	○						○	—	
④	○						—	—	



[设定数据]

设定数据	含义	数据类型
①	低限值(最小输出阈值)	BIN 16/32 位
②	高限值(最大输出阈值)	
③	高限值和低限值控制的控制输入	
④	存储由高限值和低限值控制所控制的输出值的软元件的起始软元件号	

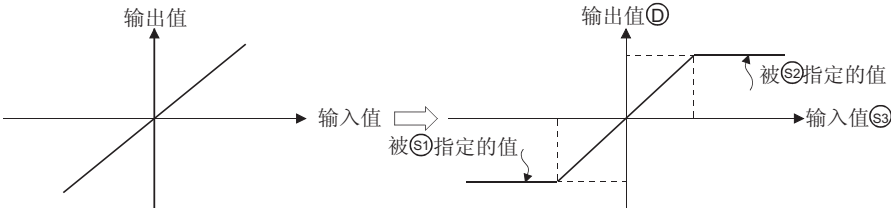
[功能]

LIMIT

(1) 检查由③指定的输入值(BIN 16 位值)，以确认其是否在由①和②指定的高限值和低限值之间，并且根据检查结果，控制在④中指定的软元件中存储的输出值。

按照下列方式控制输出值：

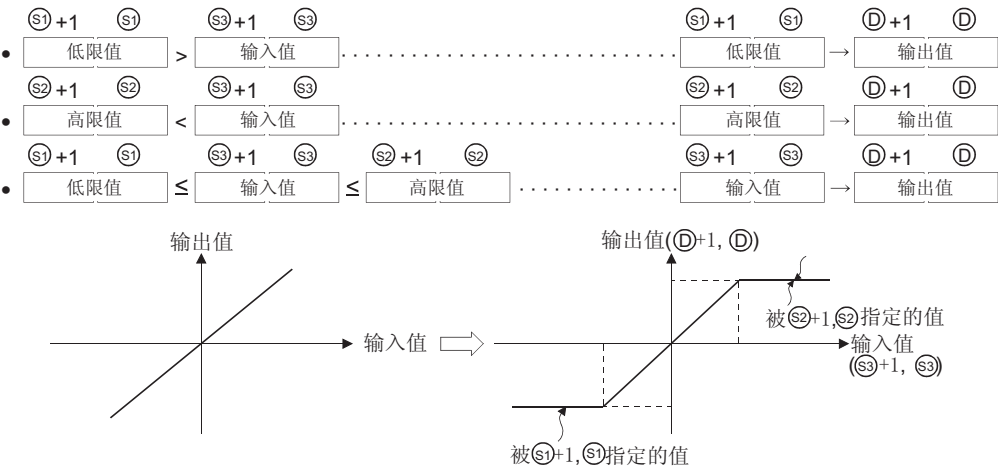
- ① 低限值 > ③ 输入值 ... ① 低限值 → ④ 输出值
- ② 高限值 < ③ 输入值 ... ② 高限值 → ④ 输出值
- ① 低限值 ≤ ③ 输入值 ≤ ② 高限值 ...
① 低限值 → ④ 输出值



- (2) 在-32768 到 32767 范围之内的值可以在(S1)、(S2)和(S3)中指定。
- (3) 当仅基于高限值执行控制时，在(S1)中指定的低限值被设置为“-32768”。
- (4) 当仅基于低限值执行控制时，在(S2)中指定的高限值设置为“32767”。

DLIMIT

(1) 根据由(D+1, D)指定的输入值(BIN 32 位值)是否在由(S3, S3+1)指定的高限值和低限值之间，控制存储在由(S2, S2+1)指定的软元件中的输出值。



- (2) 由(S1, S1+1), (S2, S2+1), 或 (S3, S3+1) 指定的值是在-2147483648 到 2147483647 的范围之内。
- (3) 仅在高限值基础上执行控制，将由(S1, S1+1)指定的低限值设置为“-2147483648”。
- (4) 仅在低限值基础执行上控制，将由(S2, S2+1)指定的高限值设置为“2147483647”。

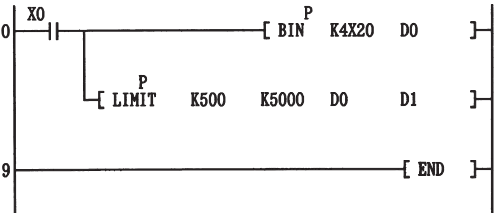
[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
 - 由(S1)指定的低限值比(S2)指定的高限值大。(错误代码：4100)

[程序示例]

(1) 当 X0 为 ON 时，下列程序对从 X20 到 X2F 的设置为 BCD 值的数据执行从 500 到 5000 的限制控制，并且在 D1 中存储结果。

[梯形图模式]



[指令模式]

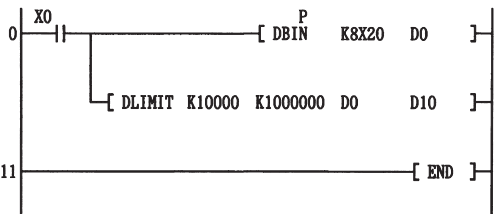
步	指令	软元件
0	LD	X0
1	BINP	K4X20 D0
4	LIMITP	K500 K5000 D0 D1
9	END	

[运算]

- 如果 $D0 < 500$ ，则 D1 变成 500。
例 $D0 = 400 \rightarrow D1 = 500$
- 当 $500 \leq D0 \leq 5000$ 时，D1 变成 D0 的值。
例 $D0 = 1300 \rightarrow D1 = 1300$
- 当 $5000 < D0$ 时，D1 变成 5000。
例 $D0 = 9600 \rightarrow D1 = 5000$

(2) 当 X0 为 ON 时，下列程序对从 X20 到 X3F 的设置为 BCD 值的数据执行从 10000 到 1000000 的限制控制。

[梯形图模式]



[指令模式]

步	指令	软元件
0	LD	X0
1	DBINP	K8X20 D0
4	DLIMIT	K10000 K1000000 D0 D10
11	END	

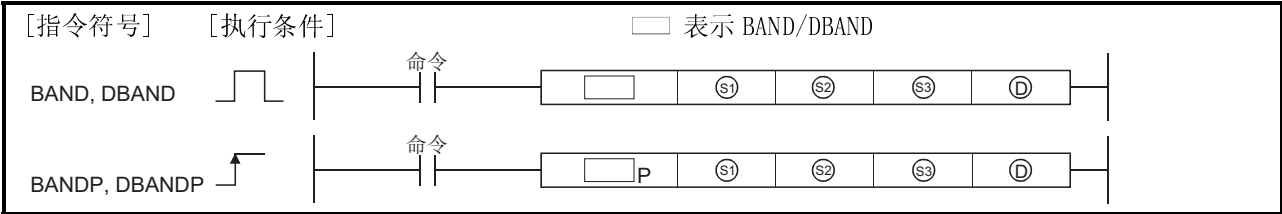
[运算]

- 如果 (D1, D0) 小于 10000，则 (D11, D10) 变成 10000。
例 $(D1, D0) = 400 \rightarrow (D11, D10) = 10000$
- 如果 $10000 \leq (D1, D0) \leq 1000000$ ，则 (D11, D10) 变成 (D1, D0) 的值。
例 $(D1, D0) = 345678 \rightarrow (D11, D10) = 345678$
- 如果 $1000000 < (D1, D0)$ ，则 (D11, D10) 变成 1000000。
例 $(D1, D0) = 9876543 \rightarrow (D11, D10) = 1000000$

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.13.2 BIN 16 位和 32 位死区控制
(BAND, BANDP, DBAND, DBANDP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[] []		特殊功能模 块 U[] \ G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
①	○						○	—	
②	○						○	—	
③	○						○	—	
④	○						—	—	



[设定数据]

设定数据	含义	数据类型
①	死区的低限值(无输出带)	BIN 16/32 位
②	死区的高限值(无输出带)	
③	基于死区控制的控制输入值	
④	存储死区控制的控制输出值的软元件的起始软元件号	

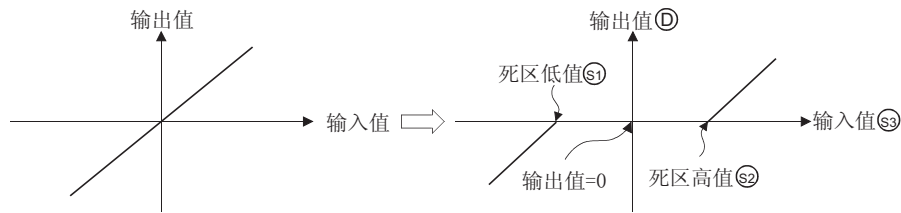
[功能]

BAND

(1) 根据由③指定的输入值(BIN 16 位值)是否在由①和②指定的高限值和低限值死区控制的范围之内，控制存储在由④指定的软元件中的输出值。

按照下列方式控制输出值：

- ① 低限值 > ③ 输入 ...
③ 输入值 — ① 低限值 → ④ 输出值
- ② 高限值 < ③ 输入值 ...
③ 输入值 — ② 高限值 → ④ 输出值
- ① 低限值 ≤ ③ 输入值 ≤ ② 高限值 ... 0 → ④ 输出值



(2) 能由 $\textcircled{S1}$, $\textcircled{S2}$, 和 $\textcircled{S3}$ 指定的值在从-32768 到 32767 的范围之内。

(3) 存储在 \textcircled{D} 中的输出值是一个带符号的 16 位 BIN 值。

因此, 如果运算结果超出了从-32768 到 32767 的范围, 将会发生下列情况:

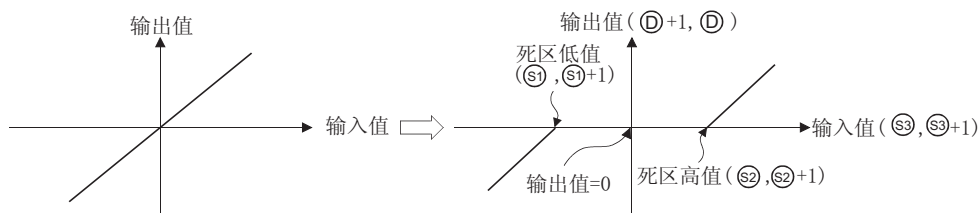
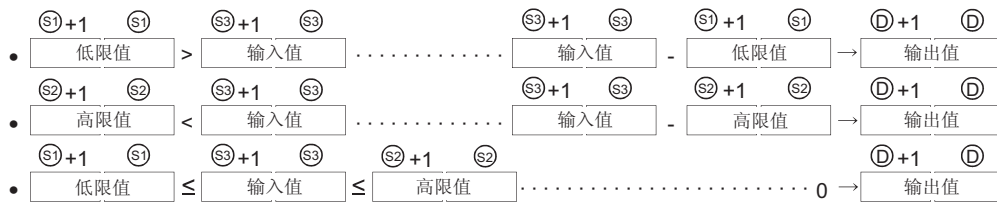
死区低限值 $\textcircled{S1}$ 10
 输入值 $\textcircled{S3}$ -32768 } 在这些情况中

$$\text{输出值} = -32768 - 10 = 8000\text{H} - \text{AH} = 7\text{FF}6\text{H} = 32758$$

DBAND

(1) 根据由 $(\textcircled{S3}, \textcircled{S3}+1)$ 指定的输入值 (BIN 32 位值) 是否由 $(\textcircled{S1}, \textcircled{S1}+1)$ 和 $(\textcircled{S2}, \textcircled{S2}+1)$ 指定的高限值和低限值死区控制范围之中, 来控制由 \textcircled{D} 指定的软元件中的输出值。

按照以下方式控制输出值:



(2) 由 $(\textcircled{S1}, \textcircled{S1}+1)$, $(\textcircled{S2}, \textcircled{S2}+1)$, 或 $(\textcircled{S3}, \textcircled{S3}+1)$ 指定的值在-2147483648 到 2147483647 的范围中。

(3) 存储在 \textcircled{D} , $\textcircled{D}+1$ 中的输出值是一个带符号的 32 位 BIN 值。因此, 如果运算结果超出了从-2147483648 到 2147483647 的范围, 将会发生下列情况:

死区低限值 $(\textcircled{S1}, \textcircled{S1}+1)$ 1000
 输入值 $(\textcircled{S3}, \textcircled{S3}+1)$ -2147483648 } 在这些情况中

$$\begin{aligned} \text{输出值} &= -2147483648 - 1000 = 80000000\text{H} - 000003\text{E}8\text{H} \\ &= 7\text{FFFFC}18\text{H} = 2147482648. \end{aligned}$$

[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 由①指定的低限值比②指定的高限值大。(错误代码：4100)

[程序示例]

- (1) 当 X0 为 ON 时，下列程序对由从 X20 到 X2F 的 BCD 值设置的数据执行从-1000 到 1000 的范围控制，并且在 D1 中存储结果。

[梯形图模式]

```
graph LR
    subgraph Ladder [ ]
        direction TB
        S0[0] -- X0 --> BIN[BIN P K4X20 D0]
        S1[1] -- BAND P K-1000 K1000 D0 D1 --> S4[4]
        S4 -- BANDP --> S9[9]
        S9 -- END --> END[END]
    end
```

[指令模式]

步	指令	软元件
0	LD	X0
1	BINP	K4X20 D0
4	BANDP	D0 K-1000 K1000 D0 D1
9	END	

[运算]

- 如果 $D0 < (-1000)$ ，则 $(D0) - (-1000)$ 的值存储在 D1 中。
例 $D0 = -2000 \rightarrow D1 = -1000$
- 当 $-1000 \leq D0 \leq 1000$ 时，值 0 存储在 D1 中。
例 $D0 = 500 \rightarrow D1 = 0$
- 如果 $1000 < D0$ ，则 $(D0) - 1000$ 的值存储在 D1 中。
例 $D0 = 7000 \rightarrow D1 = 6000$

- (2) 当 X0 为 ON 时，下列程序对在 X20 到 X3F 上设置为 BCD 值的数据执行从-10000 到 10000 的范围控制，并且在 D10 中存储结果。

[梯形图模式]

```
graph LR
    subgraph Ladder [ ]
        direction TB
        S0[0] -- X0 --> DBIN[DBIN P K8X20 D0]
        S1[1] -- DBAND P K-10000 K10000 D0 D10 --> S4[4]
        S4 -- DBANDP --> S11[11]
        S11 -- END --> END[END]
    end
```

[指令模式]

步	指令	软元件
0	LD	X0
1	DBINP	K8X20 D0
4	DBANDP	D0 K-10000 K10000 D0 D10
11	END	

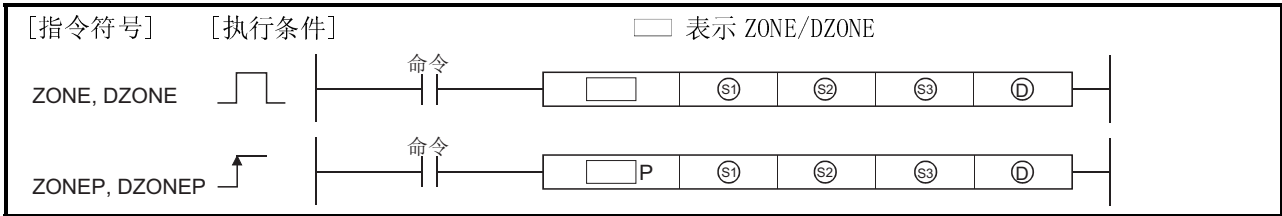
[运算]

- 如果 $(D1, D0)$ 小于 - 10000， $(D1, D0) - (-10000)$ 的值存储在 (D11, D10) 中。
例 $(D1, D0) = -12345 \rightarrow (D11, D10) = -2345$
- 如果 $-10000 \rightarrow (D1, D0) \leq 10000$ ，值 0 存储在 (D11, D10) 中。
例 $(D1, D0) = 6789 \rightarrow (D11, D10) = 0$
- 如果 $10000 < (D1, D0)$ ， $(D1, D0) - 10000$ 的值存储在 (D11, D10) 中。
例 $(D1, D0) = 50000 \rightarrow (D11, D10) = 40000$

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.13.3 BIN 16 位和 BIN 32 位数据的区域控制
(ZONE, ZONEP, DZONE, DZONEP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[] []		特殊功能模 块 U[] \ G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
①	○							○	—
②	○							○	—
③	○							○	—
④	○							—	—



[设定数据]

设定数据	含义	数据类型
①	加到输入值上的负偏差值	BIN 16/32 位
②	加到输入值上的正偏差值	
③	用于执行区域控制的输入值	
④	将要存储基于区域控制的输出值的软元件的起始软元件号	

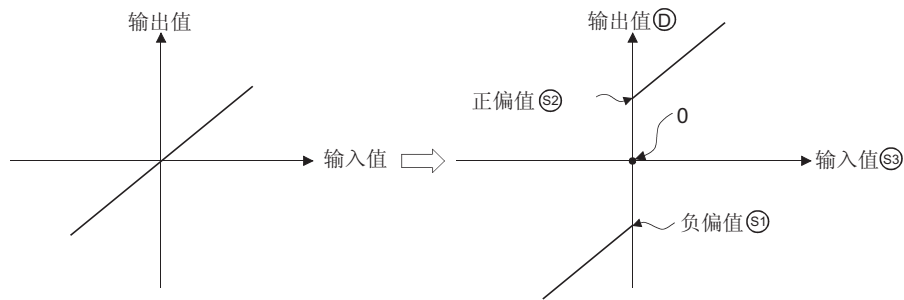
[功能]

ZONE

(1) 将①或②指定的偏差值与由③指定的输入值相加，并且在由④指定软元件号的软元件中存储结果。

按照下列方式计算偏差值：

- ③ 输入值 < 0 ...
③ 输入值 + ① 负偏差值 → ④ 输出值
- ③ 输入值 = 0 ... 0 → ④ 输出值
- ③ 输入值 > 0 ...
③ 输入值 + ② 正偏差值 → ④ 输出值



(2) 由 $\textcircled{S1}$, $\textcircled{S2}$, 和 $\textcircled{S3}$ 指定的值是在-32768 到 32767 的范围中的。

(3) 存储在 \textcircled{D} 中的输入值是带符号的 16 位 BIN 值。

因此, 如果运算结果超出了从-32768 到 32767 的范围, 将会发生下列情况:

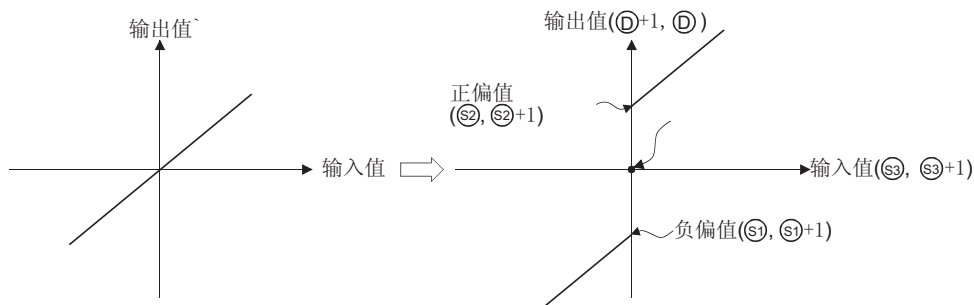
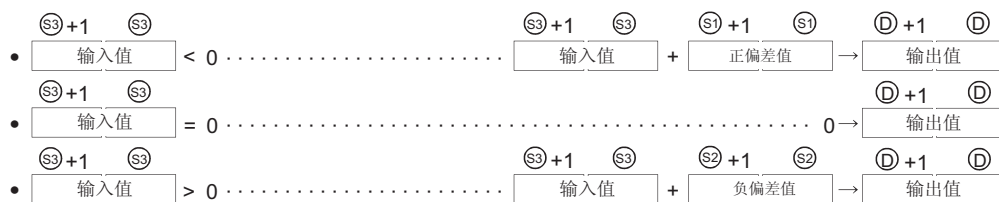
负偏差值 $\textcircled{S1}$ -100
 输入值 $\textcircled{S3}$ -32768 } 在这些情况中

$$\text{输出值} = -32768 + (-100) = 8000_{\text{H}} + \text{FF9C}_{\text{H}} = 7\text{F9C}_{\text{H}} = 32668$$

DZONE

(1) 将由 $(\textcircled{S1}, \textcircled{S1}+1)$ 或 $(\textcircled{S2}, \textcircled{S2}+1)$ 指定的偏差值与由 $(\textcircled{S3}, \textcircled{S3}+1)$ 指定的输入值相加, 并且在 $(\textcircled{D}, \textcircled{D}+1)$ 指定软元件号的软元件中存储结果。

偏差值的加法运算执行如下:



(2) 由 $(\textcircled{S1}, \textcircled{S1}+1)$, $(\textcircled{S2}, \textcircled{S2}+1)$, 或 $(\textcircled{S3}, \textcircled{S3}+1)$ 指定的值是在从-2147483648 到 2147483647 的范围之内的值。

(3) 存储在 $(\textcircled{D}, \textcircled{D}+1)$ 中的值是一个带符号的 32 位 BIN 值。因此, 如果运算结果超出了从-2147483648 到 2147483647 的范围, 将会发生下列情况

负偏差值 $(\textcircled{S1}, \textcircled{S1}+1)$ -1000
 输入值 $(\textcircled{S3}, \textcircled{S3}+1)$ -2147483648 } 在这些情况中

$$\begin{aligned} \text{输出值} &= -2147483648 + (-1000) = 80000000_{\text{H}} + \text{FFFFC18}_{\text{H}} \\ &= 7\text{FFFC18}_{\text{H}} = 2147482648. \end{aligned}$$

[运行错误]

(1) 没有与 ZONE (P) 或 DZONE (P) 指令相关的运行错误。

[程序示例]

(1) 当 X0 为 ON 时，下列程序对在 X20 到 X2F 上设置为 BCD 值的数据执行-100 到 100 的区域控制，并且在 D1 中存储结果。

[梯形图模式]		[指令模式]		
		步	指令	软元件
		0	LD	X0
		1	BINP	K4X20 D0
		4	ZONEP	K-100 K100 D0 D1
		9	END	

[运算]

- 如果 $D0 < 0$ ，则 $(D0) + (-100)$ 的值存储在 D1 中。
例 $D0 = -200 \rightarrow D1 = -300$
- 如果 $D0 = 0$ ，值 0 存储在 D1 中。
- 如果 $0 < D0$ ， $(D0) + 100$ 的值存储在 D1 中如果 $0 < D0$ 。
例 $D0 = 700 \rightarrow D1 = 800$

(2) 当 X0 为 ON 时，下列程序对在 X20 到 X3F 上设置为 BCD 值的数据执行从-10000 到 10000 的区域控制，并在 D10 中存储结果。

[梯形图模式]		[指令模式]		
		步	指令	软元件
		0	LD	X0
		1	DBINP	K8X20 D0
		4	DZONEP	K-10000 K10000 D0 D10
		11	END	

[运算]

- 如果 $(D1, D0) < 0$ ，在 $(D11, D10)$ 中存储 $(D1, D0) + (-10000)$ 的值。
例 $(D1, D0) = -12345 \rightarrow (D11, D10) = -22345$
- 如果 $(D1, D0) = 0$ ，在 $(D11, D10)$ 中存储值 0。
- 如果 $0 < (D1, D0)$ ，在 $(D11, D10)$ 中存储 $(D1, D0) + 10000$ 的值。
例 $(D1, D0) = 50000 \rightarrow (D11, D10) = 60000$

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.14 文件寄存器切换指令

7.14.1 切换文件寄存器号 (RSET, RSETP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		常数 K, H
⑤	○						

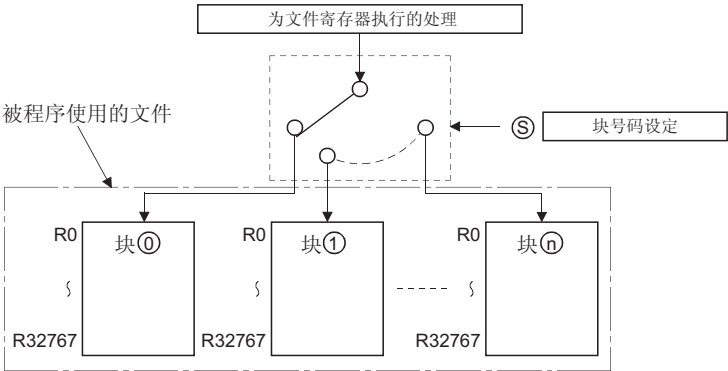
[指令符号]	[执行条件]
RSET	
RSETP	

[设定数据]

设定数据	含义	数据类型
⑤	块号码数据或存储块号码数据的软元件的软元件号	BIN 16 位

[功能]

- (1) 将程序中正在使用的文件寄存器的块号码改变为由⑤指定的软元件中存储的块号码。
在块号码改变之后，所有由顺控程序使用的文件寄存器均按照与块号码改变后的文件寄存器的关系来处理。



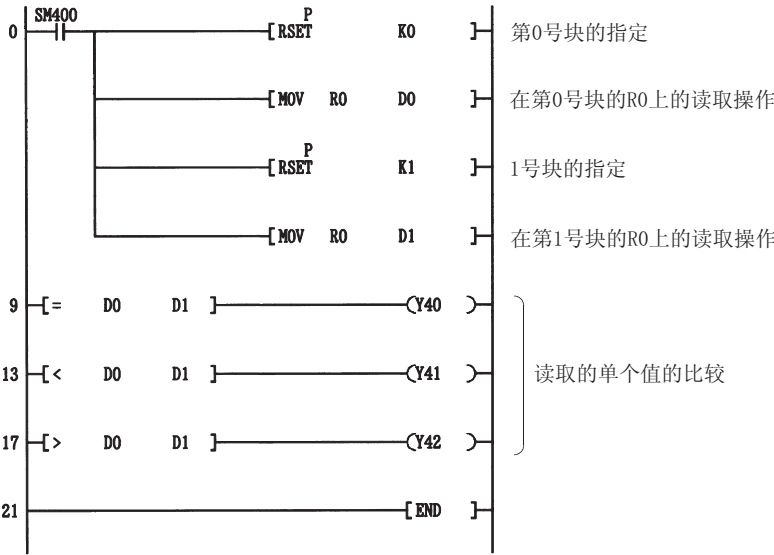
[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 由⑤指定的块号码不存在。(错误代码: 4100)
 - 由⑤指定的块号码没有文件寄存器。(错误代码: 4100)

[程序示例]

(1) 下列程序比较第 0 号块和第 1 号块的 R0。

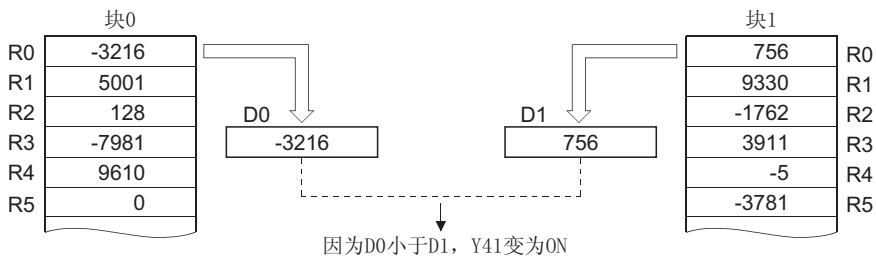
[梯形图模式]



[指令模式]

步	指令	软元件
0	LD	SM400
1	RSETP	K0
3	MOV	R0 D0
5	RSETP	K1
7	MOV	R0 D1
9	LD=	D0 D1
12	OUT	Y40
13	LD<	D0 D1
16	OUT	Y41
17	LD>	D0 D1
20	OUT	Y42
21	END	

[运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7. 14. 2 文件寄存器使用的设置文件(QDRSET, QDRSETP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		常数 \$
⑤	—	○		—	—		○

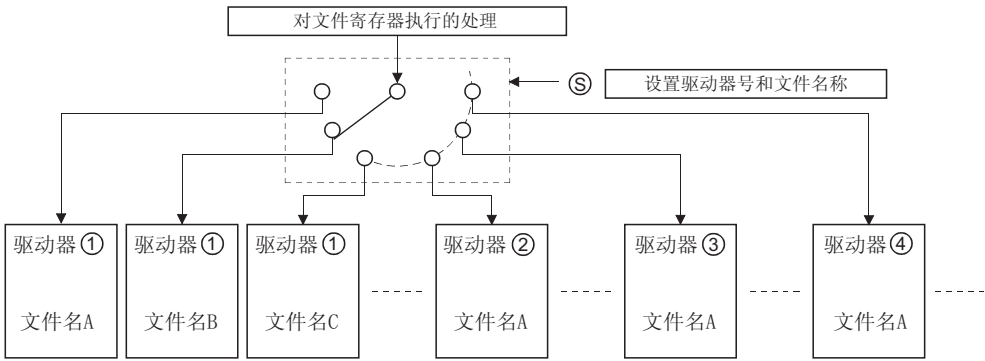
[指令符号]	[执行条件]
QDRSET	
QDRSETP	

[设定数据]

设定数据	含义	数据类型
⑤	用于将被设置的文件寄存器文件的文件名的软元件号和字符串数据，或正在存储字符串的软元件的起始软元件号。	字符串

[功能]

- (1) 将程序使用的文件寄存器的文件名改变为由⑤指定的软元件中存储的文件名。
在文件名改变之后，所有正在被顺控程序使用的文件寄存器均处理重命名的文件的第 0 号块的文件寄存器。
通过 RSET 指令执行块号码切换。



- (2) 驱动器号能够在 1 到 4 的范围内指定。
驱动器号不能指定为驱动 0 (程序内存/内部内存)
- (3) 不需要指定文件的扩展名 (.QDR)。
- (4) 通过为文件名指定 NULL 字符 (00h) 能够删除文件名设定。

(5) 即使驱动器号和文件名已经在参数中被指定，用该指令指定的文件名也将有更高的优先级。

要点
(1) 如果文件寄存器文件由 QDRSET 指令改变，将 PLC CPU 从 STOP 切换到 RUN，将程序使用的文件寄存器文件恢复为用参数设置的文件名。 要想在 PLC CPU 从 STOP 切换到 RUN 时，使用 QDRSET 指令指定的新的文件名，则在从 STOP 切换到 RUN 时，用特殊继电器 SM402 执行 QDRSET 指令。
(2) 当在下列设定中，文件寄存器被指定为刷新元件时，请不要用 QDRSET 指令改变文件寄存器的文件名。 <ul style="list-style-type: none">● MELSECNET/H 的刷新设置● CC-Link 的刷新设置● 智能化功能模块的自动刷新模式● 多 PLC 系统的自动刷新模式

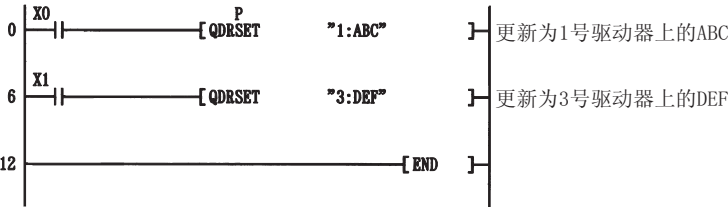
[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SDO 中。
- 在由⑤指定的驱动器号中文件名不存在。(错误代码：2410)

[程序示例]

- (1) 当 X0 为 ON 时，下列程序将目标文件切换为在 1 号驱动中的文件名为 ABC.QDR 的文件，并且当 X1 为 ON 时切换为在 3 号驱动中文件名为 DEF.QDR 的文件。

[梯形图模式]



[指令模式]

步	指令	软元件
0	LD	X0
1	QDRSETP	"1:ABC"
6	LD	X1
7	QDRSET	"3:DEF"
12	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7. 14. 3 注释的文件设置(QCDSET, QCDSETP)

设定数据	可用软元件					
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		其它
	位	字		位	字	
⑤	—	○		—	—	○

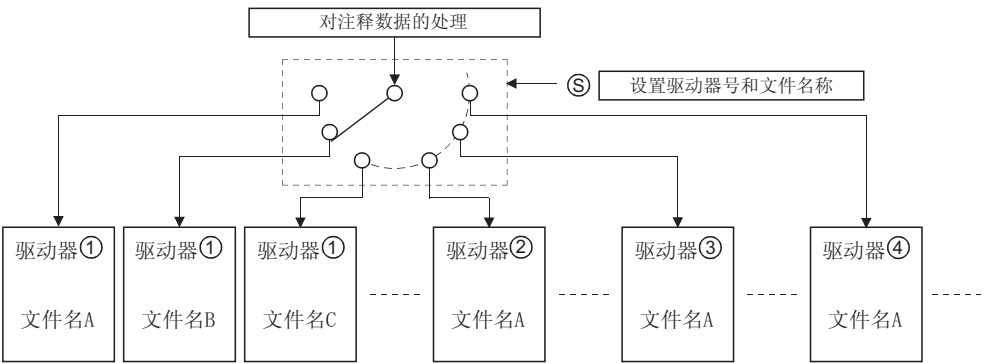


[设定数据]

设定数据	含义	数据类型
⑤	将要设置注释文件的文件名的驱动器号和字符串数据，或正在存储字符串数据的软元件的起始软元件号。	字符串

[功能]

- (1) 将由程序使用的文件寄存器文件名改变成在由⑤指定的软元件中存储的文件名。
- 文件名改变之后，正在被由顺控程序使用的注释数据按照与改变后的文件名的注释数据的关系执行处理。



- (2) 驱动器号能够在 1 到 4 的范围内指定。
- 驱动器号不能指定为驱动器 0 (程序内存/内部内存)。
- * 不能为 QCPU 指定驱动器 3。
- (3) 不需要指定文件的扩展名 (. QCD)。
- (4) 可以通过为文件名指定 NULL 字符来删除文件名设定。

(5) 即使驱动器号和文件名已经在参数指定了，用该指令指定的文件名将有更高的优先级。

[运行错误]

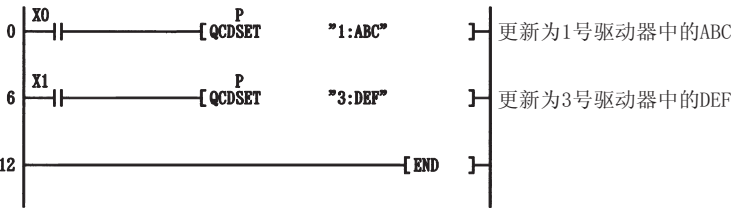
- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 在由⑤指定的驱动器号中文件名不存在。

(错误代码：2410)

[程序示例]

- (1) 当 X0 为 ON 时，下列程序将目标文件切换为 0 号驱动中文件名为 ABC.QDR 的文件，并且当 X1 为 ON 时切换为 1 号驱动中文件名为 DEF.QDR 的文件。

[梯形图模式]



[指令模式]

步	指令	软元件
0	LD	X0
1	QCDSETP	"1:ABC"
6	LD	X1
7	QCDSETP	"3:DEF"
12	END	

要点	
如果注释文件由 QCDSET 指令改变，将 PLC CPU 从 STOP 切换到 RUN，可以将程序使用的注释文件恢复为用参数设置的文件名。	
要想在 PLC CPU 从 STOP 切换到 RUN 时，使用 QCDSET 指令指定的新的文件名，则在从 STOP 切换到 RUN 时，用特殊继电器 SM402 执行 QCDSET 指令。	

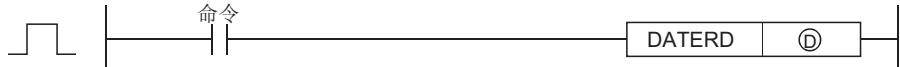
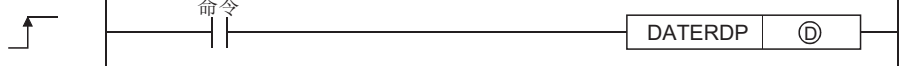
QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7. 15 时钟指令

7. 15. 1 读时钟数据 (DATERD, DATERDP)

(1) 当使用 QCPU 时

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常数	其它
	位	字		位	字				
⑤	—	○	—						

[指令符号]	[执行条件]
DATERD	
DATERDP	

[设定数据]

设定数据	含义	数据类型
①	在读取时, 存储时钟数据的软元件的起始软元件号	BIN 16 位

[功能]

(1) 从 CPU 模块的时钟软元件中读取“年, 月, 日, 分, 秒和星期几”, 并以 BIN 值存储在①指定的软元件之后。

时钟数据	⇒	①	年	(1980到2079)
		①+1	月	(1到12)
		①+2	日	(1到310)
		①+3	小时 (24小时时钟)	(0到23)
		①+4	分	(0到59)
		①+5	秒	(0到59)
		①+6	星期几	(0到6)

(2) 在①中存储的“年”是四位数表示的年。

(3) 在①+6 中存储的“星期几”为 0 到 6, 代表星期日到星期六。

星期几	星期日	星期一	星期二	星期三	星期四	星期五	星期六
存储的数据	0	1	2	3	4	5	6

(4) 闰年自动补偿。

[运行错误]

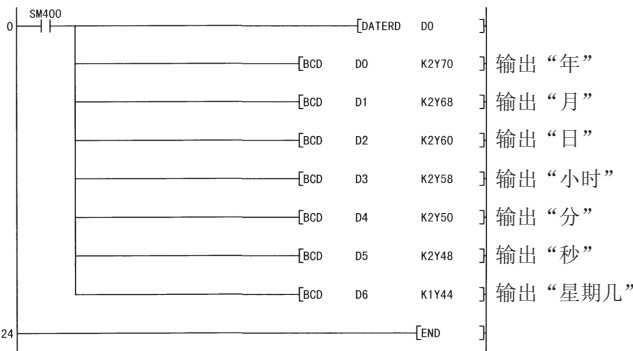
(1) 没有与 DATERD(P) 指令相关的运行错误。

[程序示例]

(1) 以下程序以 BCD 值输出下述时钟数据：

年 Y70 到 Y7F
月 Y68 到 Y6F
日 Y60 到 Y67
小时..... Y58 到 Y5F
分钟..... Y50 到 Y57
秒 Y48 到 Y4F
星期几 Y44 到 Y47

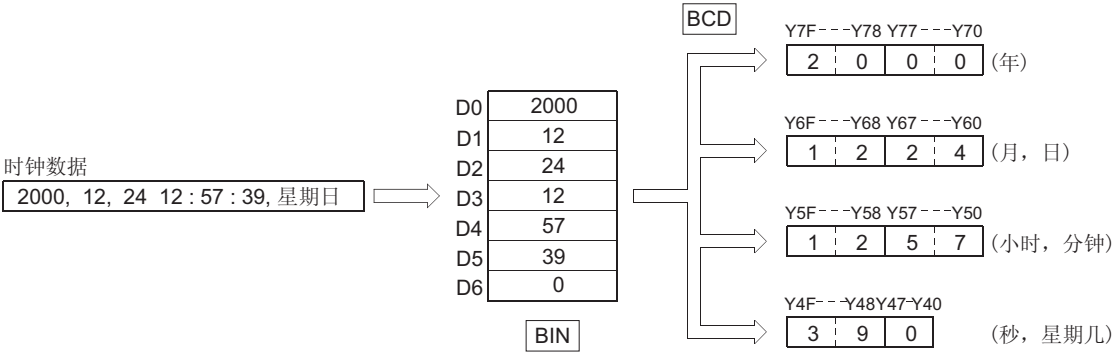
[梯形图模式]



[指令模式]

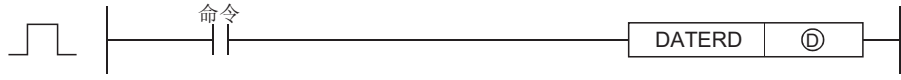
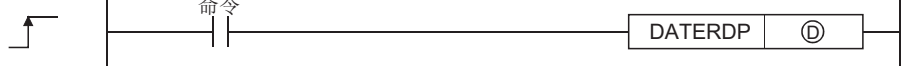
步	指令	软元件
0	LD	SM400
1	DATERD	D0
3	BCD	D0 K2Y70
6	BCD	D1 K2Y68
9	BCD	D2 K2Y60
12	BCD	D3 K2Y58
15	BCD	D4 K2Y50
18	BCD	D5 K2Y48
21	BCD	D6 K1Y44
24	END	

[运算]



(2) 当使用 QnACPU 时

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
①	—	○					—

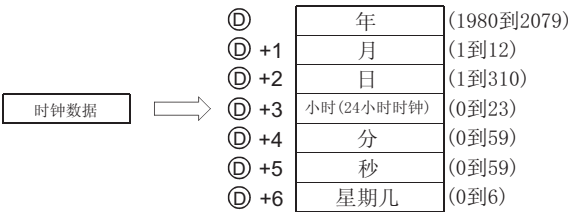
[指令符号]	[执行条件]
DATERD	
DATERDP	

[设定数据]

设定数据	含义	数据类型
①	在读取时，存储时钟数据的软元件的起始软元件号	BIN 16 位

[功能]

(1) 从 QnACPU 的时钟软元件中读取“年，月，日，分，秒和星期几”，并以 BIN 值存储在①指定的软元件之后。



(2) 在①中存储的“年”是四位数表示的年。

(3) 在①+6 中存储的“星期几”为 0 到 6，代表星期日到星期六。

星期几	星期日	星期一	星期二	星期三	星期四	星期五	星期六
存储的数据	0	1	2	3	4	5	6

(4) 闰年自动补偿。

[运行错误]

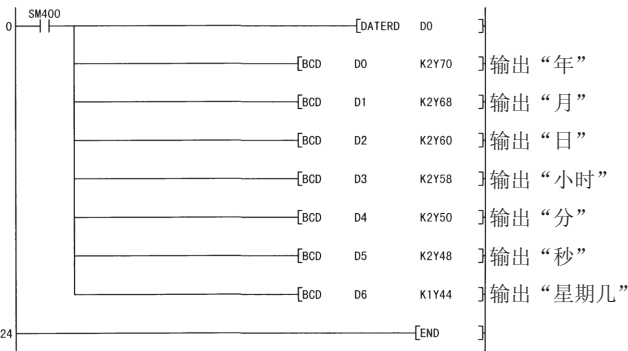
(1) 没有与 DATERD(P)指令相关的运行错误。

[程序示例]

(1) 以下程序以 BCD 值输出下述时钟数据：

年 Y70 到 Y7F
月 Y68 到 Y6F
日 Y60 到 Y67
小时..... Y58 到 Y5F
分钟..... Y50 到 Y57
秒 Y48 到 Y4F
星期几 Y44 到 Y47

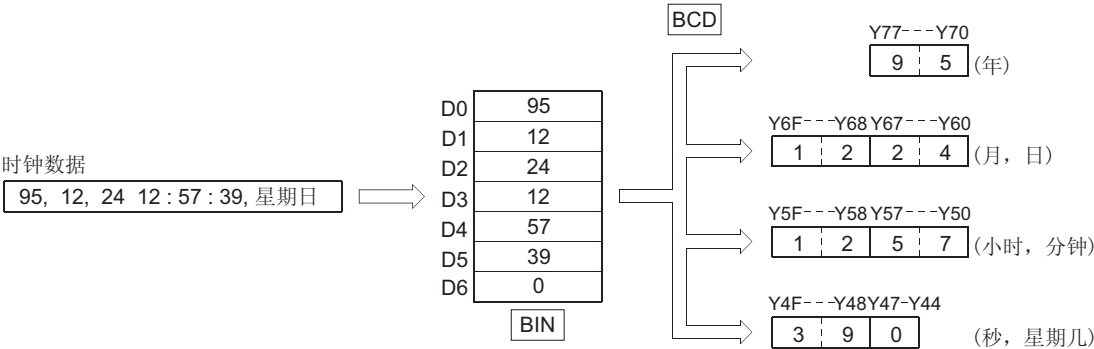
[梯形图模式]



[指令模式]

步	指令	软元件
0	LD	SM400
1	DATERD	D0
3	BCD	D0 K2Y70
6	BCD	D1 K2Y68
9	BCD	D2 K2Y60
12	BCD	D3 K2Y58
15	BCD	D4 K2Y50
18	BCD	D5 K2Y48
21	BCD	D6 K1Y44
24	END	

[运算]

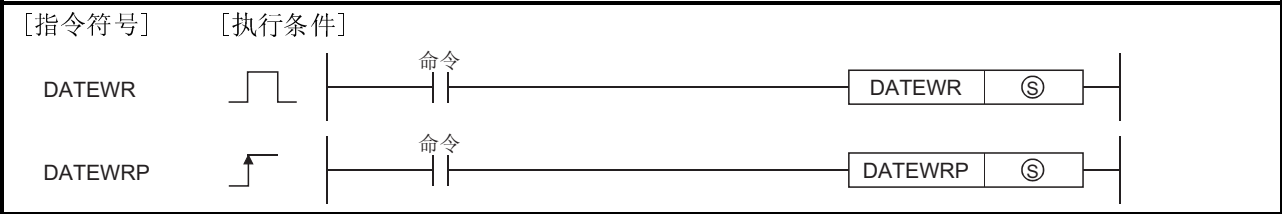


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.15.2 写时钟数据 (DATEWR, DATEWRP)

(1) 当使用 QCPU 时

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [X][Y]		特殊功能模 块 [I][O]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○					—

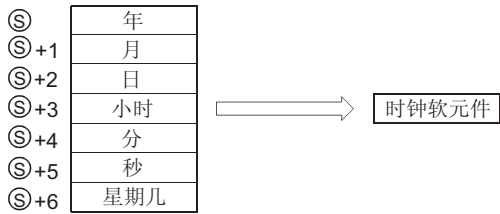


[设定数据]

设定数据	含义	数据类型
⑤	存储正要写入到时钟软元件中去的时钟数据的软元件的起始软元件号	BIN 16 位

[功能]

(1) 存储在由⑤指定号码的软元件之后的时钟数据，被写到 CPU 模块时钟软元件中



- (2) 每一个项目均被设置为 BIN 值。
- (3) 在⑤中的“年”是由在 1980 到 2079 年之间的 4 位数来表示的。
- (4) ⑤+1 指定的“月”是用从 1 到 12 的值(1 月到 12 月)来表示的。
- (5) ⑤+2 指定的“日”是用从 1 到 31 的值来表示的。
- (6) ⑤+3 指定的“小时”是用从 0 到 23 的值(使用 24 小时制时钟，从 0 小时到 23 小时)来表示的。(使用 24 小时制时钟)
- (7) ⑤+4 指定的“分钟”是用从 0 到 59 的值来表示的。
- (8) ⑤+5 指定的“秒”是用从 0 到 59 的值来表示的。

(9) S+6 指定的“星期几”是用从 0 到 6 的值(星期日到星期六)来表示的。

星期几	星期日	星期一	星期二	星期三	星期四	星期五	星期六
存储的数据	0	1	2	3	4	5	6

[运行错误]

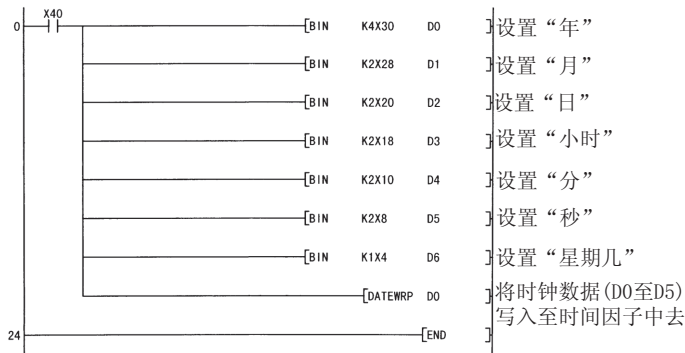
- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 数据的个别项目设置在设定范围之外。(错误代码：4100)

[程序示例]

(1) 当 X40 为 ON 时，下列程序以 BCD 值将以下时钟数据写入到时钟软元件中。

年.....	X30 到 X3F	小时	X18 到 X1F
月.....	X28 到 X2F	分钟	X10 到 X17
日.....	X20 到 X27	秒	X8 到 XF
星期几.....	X4 到 X7		

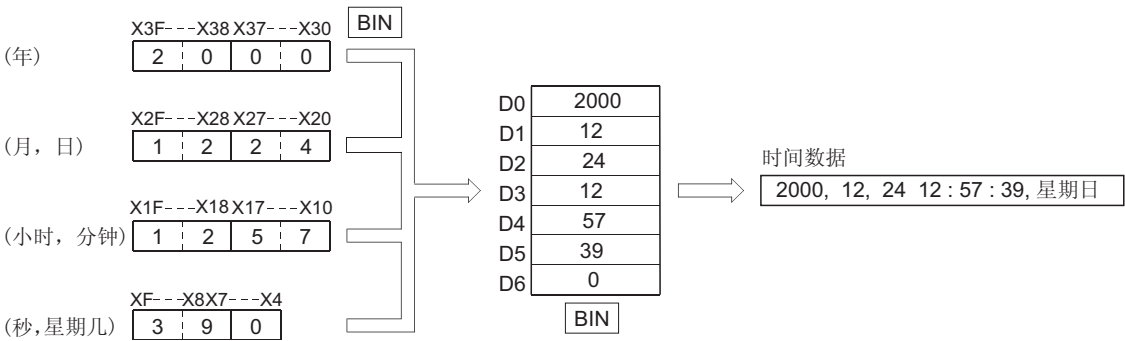
[梯形图模式]



[指令模式]

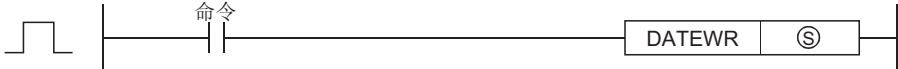
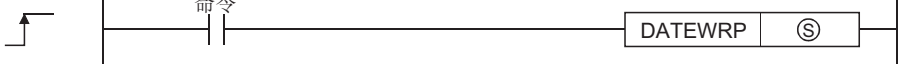
步	指令	软元件
0	LD	X40
1	BIN	K4X30 D0
4	BIN	K2X28 D1
7	BIN	K2X20 D2
10	BIN	K2X18 D3
13	BIN	K2X10 D4
16	BIN	K2X8 D5
19	BIN	K1X4 D6
22	DATEWRP	D0
24	END	

[运算]



(2) 当使用 QnACPU 时

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○					—

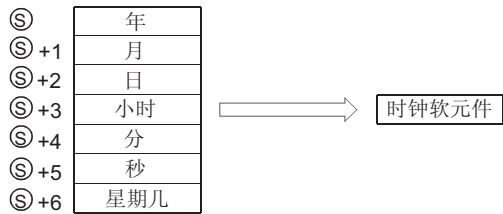
[指令符号]	[执行条件]
DATEWR	
DATEWRP	

[设定数据]

设定数据	含义	数据类型
⑤	存储将要写入到时钟软元件的时钟数据的软元件的起始软元件号	BIN 16 位

[功能]

(1) 将在由⑤指定号码的软元件之后的时钟数据写入到 CPU 模块时钟软元件中。



- (2) 每一个项目均被设置为 BIN 值。
- (3) 由⑤指定的“年”被设置为四位数的后两位数。
1995 年应设置为“95”。
- (4) ⑤+1 指定的“月”是用从 1 到 12 的值来表示的。(1 月到 12 月)
- (5) ⑤+2 指定的“日”是用从 1 到 31 的值来表示的。
- (6) ⑤+3 指定的“小时”是用从 0 到 23 的值来表示(使用 24 小时制时钟，从 0 小时到 2300 小时)(使用 24 小时制时钟)。
- (7) ⑤+4 指定的“分钟”是用从 0 到 59 的值来表示的。
- (8) ⑤+5 指定的“秒”是用从 0 到 59 的值来表示的。

(9) S+6 指定的“星期几”是用从 0 到 6 的值来表示的(星期日到星期六)。

星期几	星期日	星期一	星期二	星期三	星期四	星期五	星期六
存储的数据	0	1	2	3	4	5	6

[运行错误]

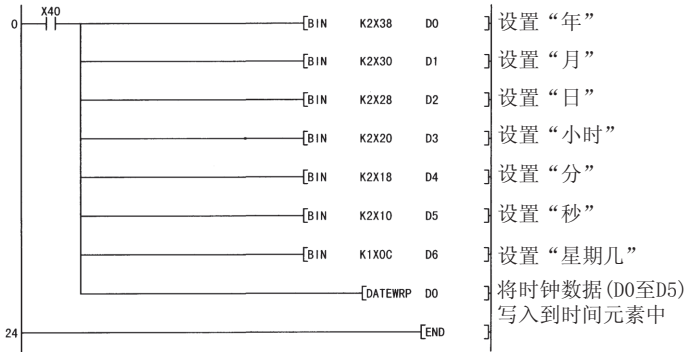
- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 数据的个别项目设置在设定范围之外。(错误代码：4100)

[程序示例]

(1) 当 X40 为 ON 时，下列程序将以下时钟数据做为 BCD 值写入到时钟软元件中去。

年.....	X38 到 X3F	小时	X20 到 X27
月.....	X30 到 X37	分钟	X18 到 X1F
日.....	X28 到 X2F	秒	X10 到 X17
星期几.....	XC 到 XF		

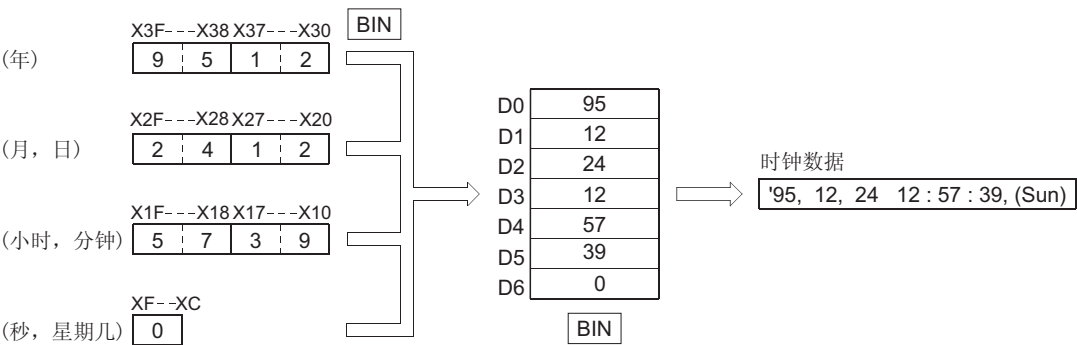
[梯形图模式]



[指令模式]

步	指令	软元件
0	LD	X40
1	BIN	K2X38 D0
4	BIN	K2X30 D1
7	BIN	K2X28 D2
10	BIN	K2X20 D3
13	BIN	K2X18 D4
16	BIN	K2X10 D5
19	BIN	K1X0C D6
22	DATEWRP	D0
24	END	

[运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.15.3 时钟数据加法运算 (DATE+, DATE+P)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[K]		特殊功能模 块 U[V]G	变址寄存器 Zn	常数	其它
	位	字		位	字				
①	—	○				—			
②	—	○				—			
③	—	○				—			

[指令符号]	[执行条件]
DATE+	
DATE+P	

[设定数据]

设定数据	含义	数据类型
①	被加的时钟数据	BIN 16 位
②	要加的时间（时钟）数据	
③	存储时钟（时间）数据加法结果的软元件的起始软元件号	

[功能]

DATE+

(1) 将存储在由①指定软元件号的软元件之后的软元件中的时间数据，加到存储在从由②指定软元件号的软元件开始的软元件中的时间数据上，并且在从由③指定软元件号的软元件开始的软元件中存储相加的结果。

①	小时	(0到23)	②	小时	(0到23)	⇒	③	小时	(0到23)
①+1	分钟	(0到59)	②+1	分钟	(0到59)		③+1	分钟	(0到59)
①+2	秒	(0到59)	②+2	秒	(0到59)		③+2	秒	(0到59)

例如：将时间 7:48:10 与 6:32:40 相加会发生以下运算：

①	小时:6	+	②	小时:7	⇒	③	小时:14
①+1	分钟:32		②+1	分钟:48		③+1	分钟:20
①+2	秒:40		②+2	秒:10		③+2	秒:50

(2) 如果时间相加的结果超过了 24 小时，将会从相加的结果中减去 24 小时，从而得到最终的运算结果。

例如：如果时间 20:20:20 与 14:20:30 相加，结果不会是 34:40:50，而是 10:40:50。

①	小时:14	+	②	小时:20	⇒	③	小时:10
①+1	分钟:20		②+1	分钟:20		③+1	分钟:40
①+2	秒:30		②+2	秒:20		③+2	秒:50

备注

关于能够设置为小时、分钟和秒的数据的更多信息，请参见章节 7.15.2。

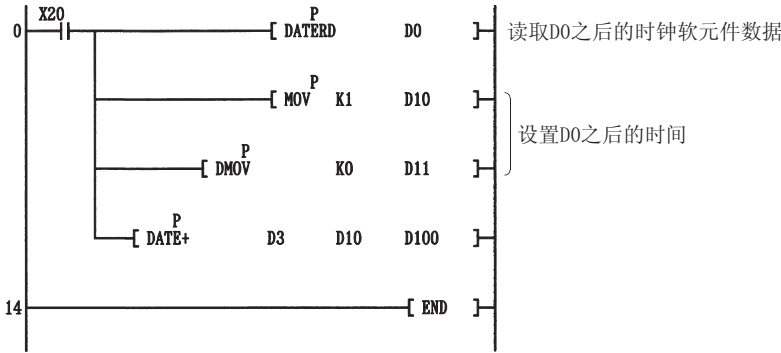
[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 由S1和S2设置的数据在设定范围之外。

[程序示例]

- (1) 当 X20 为 ON 时，下列程序将从时钟软元件中读取的时钟数据加上 1 小时，并在 D100 之后存储结果。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X20
1	DATERDP	D0
3	MOV	K1 D10
6	DMOV	K0 D11
10	DATE+P	D3 D10 D100
14	END	

[运算]

- 由 DATERDP 指令触发的时钟数据读取操作。

时钟软元件	⇒	D0	95	年
		D1	5	月
		D2	15	日
		D3	10	小时
		D4	23	分
		D5	41	秒
		D6	2	星期几

时间数据

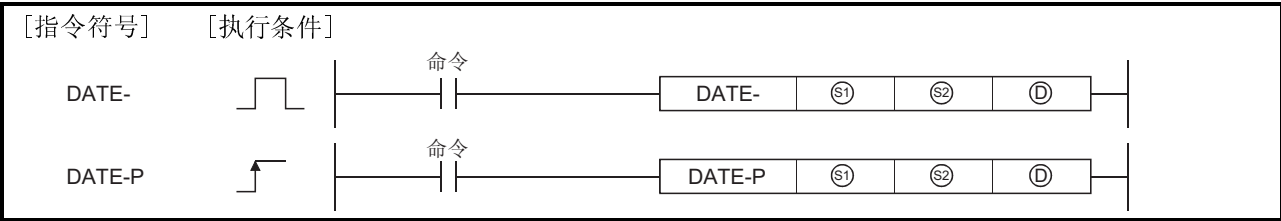
- 由 DATE+P 指令触发的加法运算。

D3	小时:10	+	D10	小时:1	⇒	D100	小时:11
D4	分钟:23		D11	分钟:0		D101	分钟:23
D5	秒:41		D12	秒:0		D102	秒:41

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.15.4 时钟数据减法运算 (DATE-, DATE-P)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
①	—	○				—	
②	—	○				—	
③	—	○				—	



[设定数据]

设定数据	含义	数据类型
①	存储要减的时间数据的软元件的起始软元件号	BIN 16 位
②	存储要被减的时间（时钟时间）数据的软元件的起始软元件号	
③	存储减法运算的结果的软元件的起始软元件号	

[功能]

(1) 从在由①指定软元件号的软元件中存储的时间数据中，减去从由②指定软元件号的软元件开始的软元件中存储的时间数据，并且在从由③指定软元件号的软元件开始的软元件中存储减法运算结果。

①	小时	(0到23)	②	小时	(0到23)	⇒	③	小时	(0到23)
①+1	分钟	(0到59)	-	②+1	分钟		③+1	分钟	(0到59)
①+2	秒	(0到59)		②+2	秒		③+2	秒	(0到59)

例如：如果从时钟时间 10:40:20 中减去时钟时间 3:50:10，运算执行如下：

①	小时:10	(0到23)	②	小时:3	(0到23)	⇒	③	小时:6	(0到23)
①+1	分钟:40	(0到59)	-	②+1	分钟:50		③+1	分钟:50	(0到59)
①+2	秒:20	(0到59)		②+2	秒:10		③+2	秒:10	(0到59)

(2) 如果减法运算的结果是一个负数，那么将在结果上加上 24，从而得到最终的运算结果。
例如：如果从 4:50:32 中减去时钟时间 10:42:12，结果不会是-6:8:20，而是 18:8:20。

Ⓢ1	小时:4	-	Ⓢ2	小时:10	⇒	Ⓢ	小时:18
Ⓢ1+1	分钟:50		Ⓢ2+1	分钟:42		Ⓢ+1	分钟:8
Ⓢ1+2	秒:32		Ⓢ2+2	秒:12		Ⓢ+2	秒:20

备注

关于能够设置为小时，分钟和秒的数据的更多信息，请参见章节 7.15.2。

[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 由Ⓢ1和Ⓢ2设置的数据在设定范围之外。

[程序示例]

- (1) 当 X1C 为 ON 时，下列程序从时钟软元件中读取的时钟数据中减去从 D10 开始的软元件中存储的时间数据，并且在从 R10 开始的软元件中存储结果。

[梯形图模式]		[列表模式]	
	步	指令	软元件
	0	LD	X1C
	1	DATERDP	D100
	3	DATE-P	D103 D10 R10
	7	END	

[运算]

- 由 DATERDP 触发的时间数据读取运算。

时钟软元件	⇒	D100	95	年
		D101	4	月
		D102	20	日
		D103	3	小时
		D104	21	分
		D105	20	秒
		D106	1	星期几

时间数据

- 由 DATE-P 指令触发的减法运算（当 D10 到 D12 已经指定了 10 小时，40 分钟，和 10 秒时）

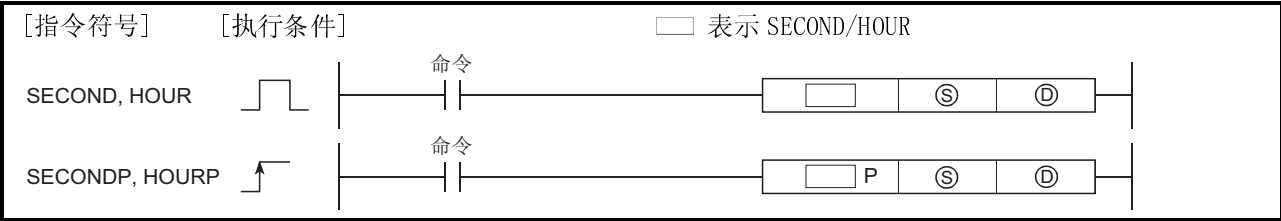
D103	小时:3		D10	小时:10		R10	小时:16
D104	分钟:21	-	D11	分钟:40	⇒	R11	分钟:41
D105	秒:20		D12	秒:10		R12	秒:10
3:21:20 - 10:40:10		⇒	-8:41:10		⇒	16:41:10	
			↓				
			用24加到该值上				

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.15.5 改变时间数据格式(SECOND, SECONDP, HOUR, HOURP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		常数 K, H
⑤*1	—	○			—		—
⑥*1	○	○			○		—
⑤*2	○	○			○		○
⑥*2	—	○			—		—

*1: 用于 SECOND(P) 指令 *2: 用于 HOUR(P) 指令



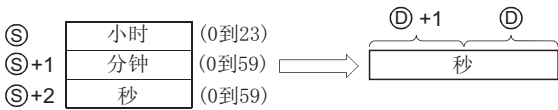
[设定数据]

设定数据	含义	数据类型
⑤	存储改变之前的时钟数据的软元件的起始软元件号	BIN 16/32 位
⑥	存储改变之后的时钟数据的软元件的起始软元件号	BIN 16/32 位

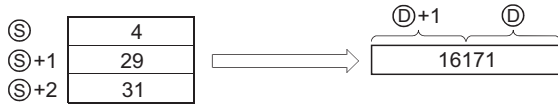
[功能]

SECOND

(1) 将存储在由⑤指定的软元件之后的时间数据转换为秒，并且在由⑥指定的软元件中存储转换结果。



例如，如果值为 4 小时，29 分钟，31 秒，转换执行如下：



HOUR

(1) 将由⑤指定软元件号的软元件中存储的秒数据转换为小时，分钟和秒，并且在由⑥指定的软元件之后存储运算结果。



例如，如果被指定的值为 45325 秒，执行的转换运算如下：

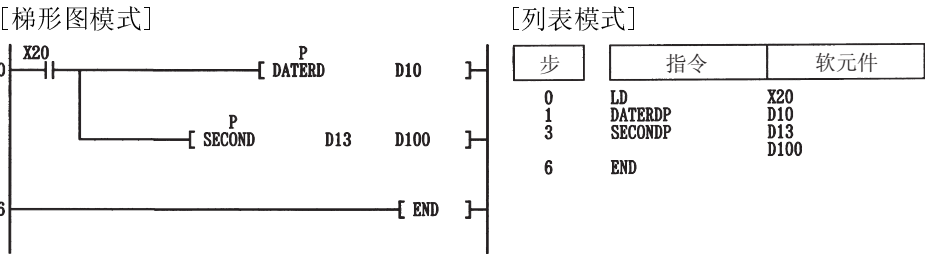


[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 由 S 指定的数据在可接受的范围之外。 (错误代码：4100)

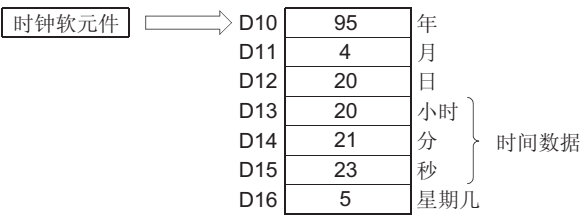
[程序示例]

- (1) 当 X20 变为 ON 时，下列程序将从时钟软元件中读取的时钟时间数据转换为秒，并且在 D100 和 D101 中存储结果。

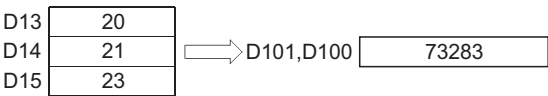


[运算]

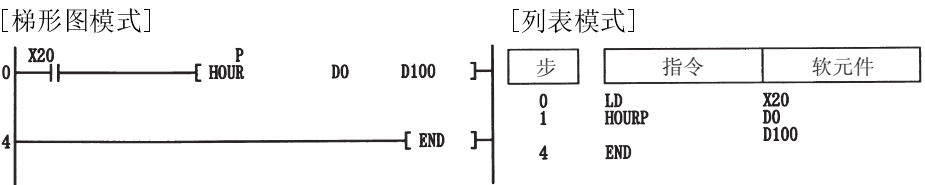
- 由 DATERDP 触发的时间数据读取运算。



- 当由 SECONDP 指令触发时转换为秒。

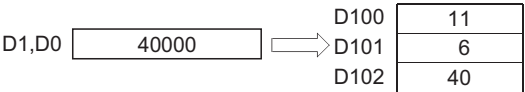


- (2) 当 X20 为 ON 时，下列程序将存储在 D0 和 D1 中的秒转换为小时、分钟、秒格式，并且在从 D100 开始的软元件中存储结果。



[运算]

- 当由 HOURP 指令触发时转换为小时、分钟和秒格式（当 D1 和 D0 指定的值为 40000 秒时）



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	×	×	○	○

7.16 外围设备指令

7.16.1 将信息显示到外围设备 (MSG)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常数 \$	其它
	位	字		位	字				
⑤	—	○	—				○	—	

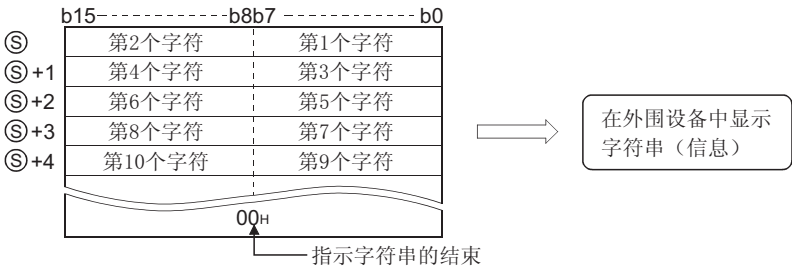


[设定数据]

设定数据	含义	数据类型
⑤	要显示的字符串数据, 或存储字符串数据的软元件的起始软元件号	字符串

[功能]

- (1) 将从由⑤指定软元件号的软元件开始的软元件中存储的字符串数据, 作为一条信息显示到在终端模式中设置的外围设备中。
- 该字符串数据的结束位置用 00H 作为标记。



- (2) 在外围设备中显示的字符串数据最多能包括 64 个字符。
- (3) 由⑤指定的字符串数据存储从 SD738 到 SD769 的位置上 (信息存储区域)。
- (4) 当执行 MSG 指令时, SM738 (MSG 指令执行信号) 被变为 ON。
- 当 SM738 变为 ON 时, 将不会处理任何其它被执行的 MSG 指令。
- (5) 当在外围设备中显示字符串 (信息) 时, SM738 变为 OFF, 并且从 SD738 到 SD769 上存储的字符串数据被清除 (00H)。

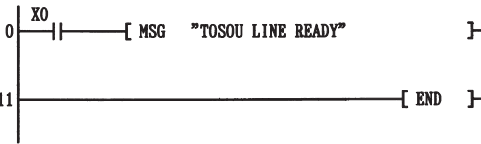
[运行错误]

(1) 没有与 MSG 指令相关的运行出错。

[程序示例]

(1) 当 X0 变为 ON 时，下列程序传送信息 “TOSOU LINE READY” 以将其显示在一个外围设备上。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	MSG	"TOSOU LINE READY"
11	END	

[运算]

	b15-----b8	b7-----b0
SD738	4F _H (O)	54 _H (T)
SD739	4F _H (O)	53 _H (S)
SD740	20 _H (空格)	55 _H (U)
SD741	49 _H (I)	4C _H (L)
SD742	45 _H (E)	4E _H (N)
SD743	52 _H (R)	20 _H (空格)
SD744	41 _H (A)	45 _H (E)
SD745	59 _H (Y)	44 _H (D)
SD746	00 _H	

显示 →

备注

1) 关于讨论的如何在终端模式中操作外围设备，请参考外围设备操作手册。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	×	×	○	○

7.16.2 外围设备的键盘输入(PKEY)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 [J][K]		特殊功能模 块 [U][G]	变址寄存器 Zn
	位	字		位	字		
①	—	○					—



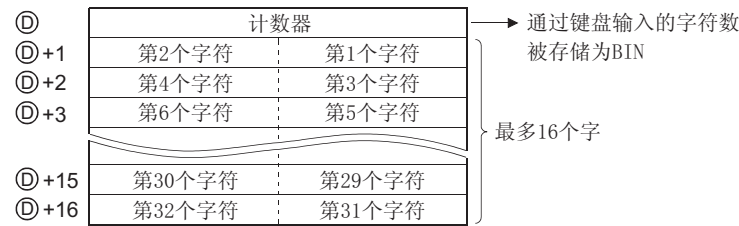
[设定数据]

设定数据	含义	数据类型
①	将要存储键盘输入的软元件的起始软元件号	BIN 16 位

[功能]

- (1) 在执行 PKEY 指令时，17 个字被从由①指定的软元件中清除，并且 SM736 (PKEY 指令执行中标记) 变为 ON。
另外，SM737 (键盘输入接收标记) 变为 OFF。
当 PKEY 指令执行结束时，在由①指定的软元件之后，在终端模式中被指定的外围设备中的键盘输入数据被存储为 ASCII 码。
- (2) 当 PKEY 指令变为 OFF 时，SM736 和 SM737 均变为 OFF。
- (3) 当键盘输入的一个字符被从外围设备中接收到时，SM737 变为 ON，并且当键盘输入的数据被 QnACPU 存储时，SM737 变为 OFF。
然而当 SM737 是 ON 时，来自外围设备的键盘输入将不会被接受。
- (4) 当从外围设备中键入“CR”时，“CR”的 ASCII 码 (0Dh) 将附加在键入数据的末尾，从而结束从外围设备的键入操作。
- (5) 能够从外围设备中输入的数据量为 32 个字符。
如果从外围设备中已经输入了 32 个字符，即使可能没有输入“CR”，来自外围设备的键盘输入也会结束。

(6) 对由⑤指定的软元件之后的数据，存储执行如下：



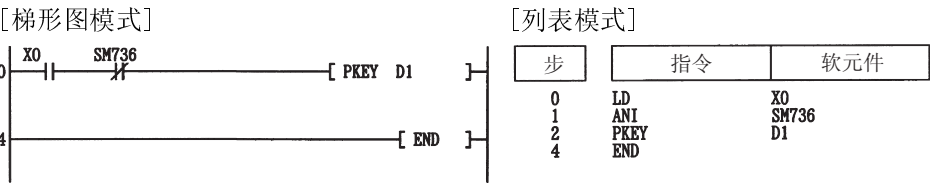
(7) PKEY 指令不能在两个或更多的位置上同时执行。
如果需要在两个或更多的位置上使用 PKEY 指令，可以用 SM736 (PKEY 指令执行中标记) 来建立一个互锁，从而使得它们可以同时被执行。

[运行错误]

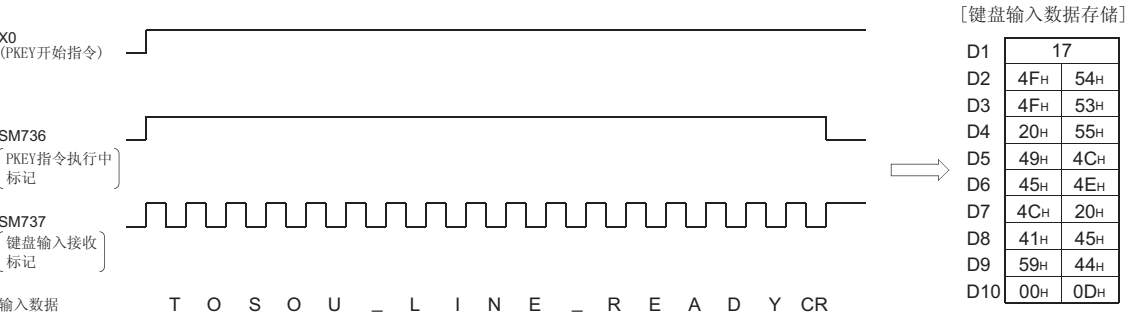
- (1) 在下列发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 试图去存储超过由⑤指定的软元件范围的键盘输入数据。 (错误代码：4101)

[程序示例]

(1) 在 X0 变为 ON 后，下列程序在 D1 之后从外围设备中输入字符 “TOSOU LINE READY”。



[运算]



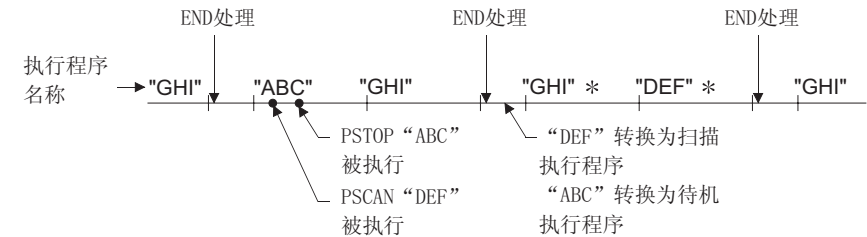
QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.17 程序控制指令

(1) 当执行类型被程序控制指令转换时的处理如下：

执行的指令 改变之前的 的执行类型	PSCAN	PSTOP	POFF	PLOW
扫描执行类型	不改变-保持扫描类型执行	变为待机类型	在下一个扫描中输出被变为 OFF，在这之后的下一个扫描开始变成待机类型	变成低速执行类型
初始执行类型				
待机类型	变为扫描执行类型	不改变-保持待机类型	无处理	
低速执行类型	低速执行类型的执行被停止，从下一个扫描开始变为扫描执行类型。 (从 0 步开始执行)	低速执行类型的执行被停止，从下一个扫描开始变为待机类型	低速执行类型的执行被停止，在下一个扫描中输出变为 OFF。在这之后的下一个扫描开始变成待机类型。	不改变-保持低速执行类型

(2) 因为 PSCAN 和 PSTOP 指令执行的程序执行类型转换是出现在 END 处理的时候，所以这类转换不可能出现在程序执行中。当在同一个扫描中出现了为同一个程序设置了不同的执行类型时，那么执行类型就是最后执行的执行转换指令指定的那个执行类型。



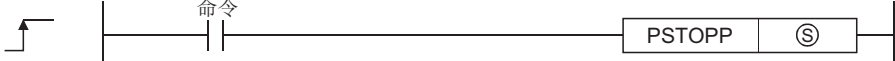
备注

- 1) *： “GHI” 和 “DEF” 程序执行的顺序是由程序设定参数决定的。
- (3) 当执行 POFF 指令时，在下一个扫描时输出变为 OFF，并且在第三个扫描和后续扫描中执行类型将变为待机类型。
如果是在输出 OFF 处理之前被执行，则程序控制指令将被忽略。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.17.1 程序待机指令（PSTOP, PSTOPP）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]K[]		特殊功能模 块 U[]V[]G[]	变址寄存器 Zn	常数 \$	其它
	位	字		位	字				
⑤	—	○		—				○	—

[指令符号]	[执行条件]
PSTOP	
PSTOPP	

[设定数据]

设定数据	含义	数据类型
⑤	将要设置为待机状态的程序文件名的字符串数据，或存储字符串数据的软元件的起始软元件号。	字符串

[功能]

- (1) 将存储在由⑤指定的软元件中的文件名程序设置为待机状态。
- (2) 只有存储在 0 号驱动器(程序内存/内部 RAM)中的程序才能被设置为待机类型。
- (3) 当 END 处理被执行时，指定的程序被设置为待机状态。
- (4) 即使程序执行类型已经在参数中指定，该指令也将有更高的优先级。
- (5) 不需要指定文件名的扩展名(.QPG)。(只有.QPG 文件才能进行操作)

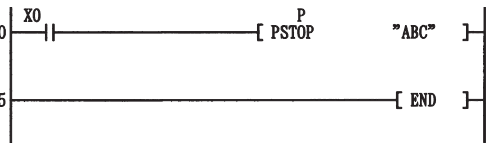
[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
 - 指定文件名的程序不存在。(错误代码: 2410)

[程序示例]

- (1) 当 X0 变为 ON 时，下列程序将文件名为 ABC 的程序设置到待机状态中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	PSTOPP	"ABC"
5	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.17.2 程序输出 OFF 待机指令(POFF, POFFP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数 \$	其它
	位	字		位	字				
⑤	—	○	—				○	—	

[指令符号]	[执行条件]
POFF	
POFFP	

[设定数据]

设定数据	含义	数据类型
⑤	变为 OFF 输出来指定待机状态的程序的文件名, 或存储文件名的软元件。	字符串

[功能]

- (1) 改变文件名存储在由⑤指定的软元件中的程序的执行类型。
 - 扫描执行类型：在下一个扫描中变为 OFF 输出（无执行处理）。在后续扫描中程序被设置为待机类型。
 - 低速执行类型：停止低速执行类型的执行，并在下一个扫描变为 OFF 输出。程序在下一个扫描之后的扫描中被设置为待机类型。
- (2) 只有存储在 0 号驱动器（程序内存/内部 RAM）中的程序能够被设置为待机类型。
- (3) 即使程序执行类型已经在参数中指定，该指令也将有更高的优先级。
- (4) 不需要指定文件名的扩展名(.QPG)。（只有.QPG 文件才能进行操作。）

[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
 - 指定文件名的程序不存在。（错误代码：2410）

备注

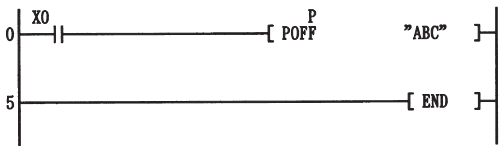
- (1) 非执行处理是指每个线圈指令的处理和条件设置为 OFF 状态时的处理是一样的。
- (2) 不管条件触点是 ON 还是 OFF，在非执行处理之后的每个线圈指令的运算结果如下：

- OUT指令..... 强制为OFF
 - SET指令
 - RST指令
 - SFT指令
 - 基本指令
 - 应用指令
 - PLS指令
 - 脉冲转换指令
(P)
 - 低速/高速定时器的
当前值..... 0
 - 累计定时器的当前值
 - 计数器的当前值
- 状态被保持
- 同条件触点为OFF时的处理一致
- 被保持

[程序示例]

- (1) 当 X0 变为 ON 时，下列程序将文件名为 ABC 的程序设置为不可执行并且将其放置在待机状态中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	POFFP	"ABC"
5	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.17.3 程序扫描执行登记指令(PSCAN, PSCANP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○		—			○

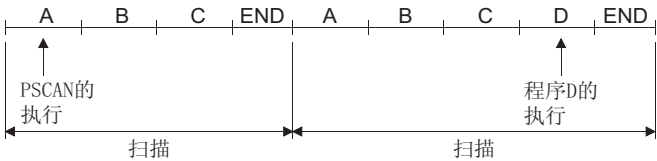
[指令符号]	[执行条件]
PSCAN	
PSCANP	

[设定数据]

设定数据	含义	数据类型
⑤	将要被置为扫描执行类型的程序的文件名，或存储文件名的软元件的起始软元件号。	字符串

[功能]

- (1) 将文件名存储在由⑤指定的软元件中的程序设置为扫描执行类型。
- (2) 只有存储在 0 号驱动器（程序内存/内部 RAM）中的程序能够被设置为扫描执行类型。
- (3) 指定的程序假定扫描执行类型带有 END 处理。
(例) 当存在程序 A, B 和 C, 并且程序 A 执行程序 D 的“PSCAN”。



- (4) 即使程序执行类型已经在参数中指定，该指令也将有更高的优先级。
- (5) 不需要指定文件名的扩展名(.QPG)。(只有.QPG 文件才能进行操作。)

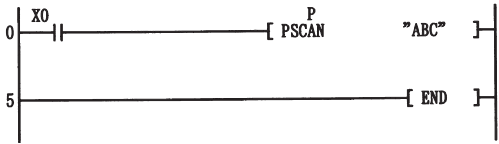
[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
 - 指定文件名的程序不存在。(错误代码: 2410)

[程序示例]

(1) 当 X0 变为 ON 时，下列程序将文件名为 ABC 的程序设置到扫描执行状态中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	PSCANP	"ABC"
5	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.17.4 程序低速执行登记指令(PLOW, PLOWP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 [J][N][C]		特殊功能模 块 [U][V][G][C]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○		—			○

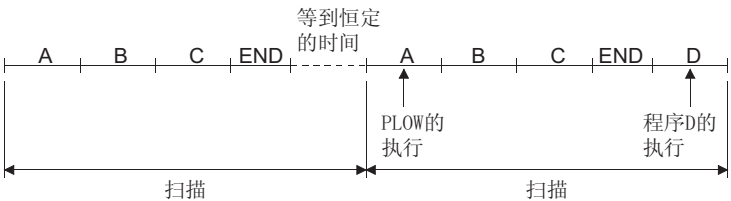
[指令符号]	[执行条件]
PLOW	
PLOWP	

[设定数据]

设定数据	含义	数据类型
⑤	要指定为低速执行类型的程序的文件名, 或存储文件名的软元件的起始软元件号	字符串

[功能]

- (1) 将文件名存储在由⑤指定的软元件中的程序设置为低速执行类型。
- (2) 只有存储在 0 号驱动器（程序内存/内部 RAM）中的程序能够被设置为低速执行类型。
- (3) 指定的程序假定低速执行类型的程序带有 END 处理。
(例)
当存在程序 A, B 和 C, 并且程序 A 执行程序 D 的“PLOW”。
(假定恒定扫描已经被设置)



- (4) 即使程序执行类型已经在参数中指定, 该指令也将有更高的优先级。
- (5) 不需要指定文件名的扩展名(.QPG)。(只有.QPG 文件才能进行操作。)

[运行错误]

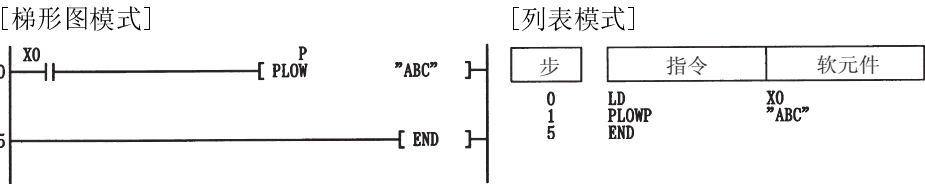
- (1) 在下列发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 指定文件名的程序不存在。

(错误代码: 2410)
- 在文件名已经被指定的程序中有一个 CHK 指令。

(错误代码: 4235)

[程序示例]

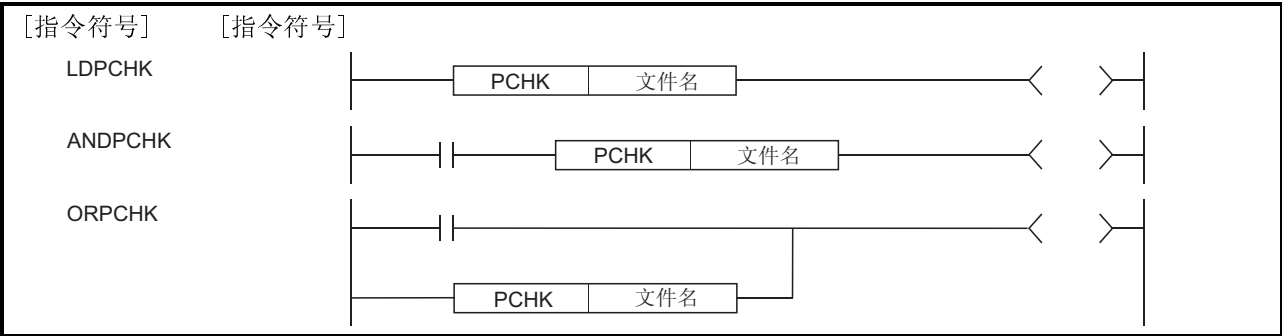
(1) 当 X0 变为 ON 时，下列程序将文件名为 ABC 的程序设置到低速执行状态中。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.17.5 程序执行状态检查指令(PCHK)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G][]	变址寄存器 Zn	常数 \$	其它
	位	字		位	字				
⑤	—						○	—	



[设定数据]

设定数据	含义	数据类型
⑤	执行状态将要被检查的程序的文件名	字符串

[功能]

- (1) 检查指定文件名的程序是否在执行。
- (2) 当指定文件名的程序在执行时该指令导通，当程序为非执行时该指令不导通。
- (3) 指定文件名时不需要指定扩展名(. QPG)。
例如，当文件名为 ABC. QPG 时，指定 “ABC” 。

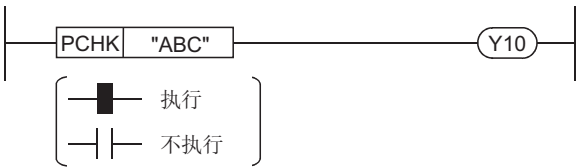
[运行错误]

在下列发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。

- 指定文件名的程序不存在。(错误代码：2410)

[程序示例]

- (1) 当执行程序文件 “ABC. QPG” 时，该程序保持 Y10 为 ON。



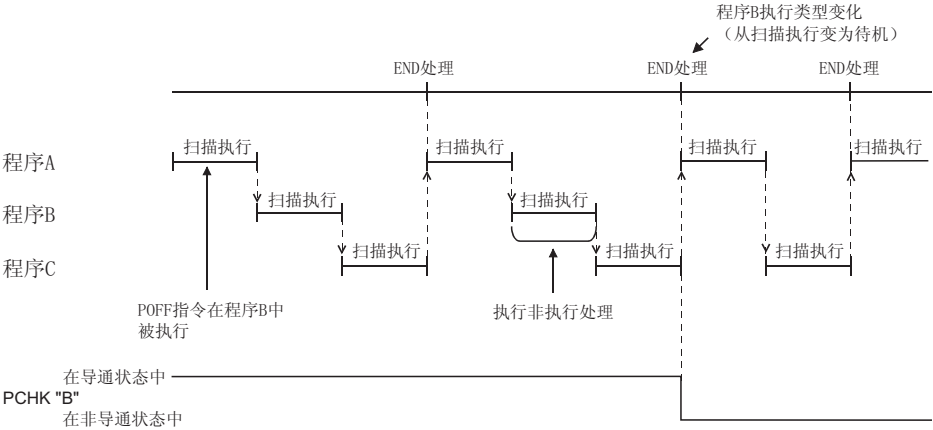
备注

“非执行”表示程序执行类型为待机类型。
“执行”表示程序执行类型为扫描执行类型（包括输出 OFF（在非执行处理过程中）的过程），低速执行类型或恒定固定扫描执行类型。

要点

当指定文件名的程序（目标程序）在执行中时，PCHK 指令处于导通状态，并且当程序是非执行时，该指令也是处于非导通状态。
当目标程序被用 POFF 指令设置为非执行（待机类型）时，在目标程序的非执行处理被执行过程中，PCHK 指令处于导通状态。
当在非执行处理结束处的扫描周期的 END 处理时，目标程序被设置为非执行（待机类型），并且 PCHK 指令被置为非导通状态。
因此请注意，当 PCHK 指令在非执行处理已经被 POFF 指令结束的程序中被执行，PCKK 指令可能被置为导通状态。

当程序 A 执行程序 B 的 POFF 指令并且程序 C 执行程序 B 的 PCHK 指令时，下列图表表示所执行的运算，其中程序是按程序 A，程序 B 和程序 C 的顺序来执行的。

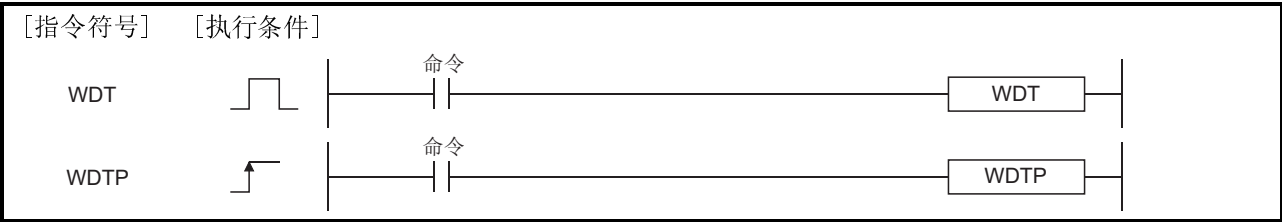


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.18 其它指令

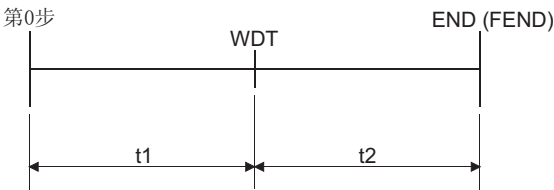
7.18.1 复位看门狗定时器(WDT, WDTP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][G[]	变址寄存器 Zn	常数	其它
	位	字		位	字				
—	—								



[功能]

- (1) 在顺控程序的执行过程中复位看门狗定时器。
- (2) 该指令用在由于主要的条件，扫描时间超过为看门狗定时器设置的值的情况中。
如果在每个扫描中，扫描时间均超过了看门狗定时器设定值，那么在外围设备设定中改变看门狗定时器的设定。
- (3) 注意既不是从 0 步到 WDT 指令的 t1，也不是从 WDT 到 END (FEND) 指令的 t2 超过了看门狗定时器的设定值。



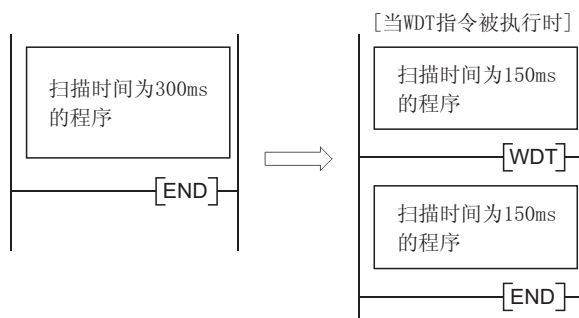
- (4) 在一个扫描中 WDT 指令能够被使用两次或更多次，但是，因为在产生错误的过程中直到输出变为 OFF 所需要的时间的原因，在这种情况下应该格外注意。
- (5) 即使 WDT 或 WDTP 指令被执行，存储在特殊寄存器中的扫描时间值也不会被清除。
相应的，当为特殊寄存器设置的扫描时间的值大于用参数设置的看门狗定时器的值时，也有这种情况发生。

[运行错误]

- (1) 没有与 WDT (P) 指令相关的运行错误。

[程序示例]

- (1) 下列程序设定了 200ms 的看门狗定时器从 0 步到 END (FEND) 指令的程序执行基于执行条件需要 300ms。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.18.2 定时脉冲发生(DUTY)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]G[]		特殊功能模 块 U[]G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
n1	○				○				—
n2	○				○				—
⑤	○*				—				—

※:只能在从 SM420 到 SM424 和从 SM430 到 SM434 中使用

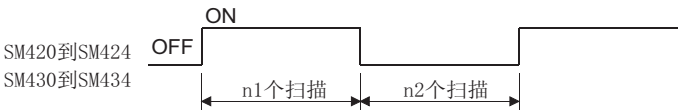
[指令符号]	[执行条件]
DUTY	

[设定数据]

设定数据	含义	数据类型
n1	强制为 ON 的扫描个数	BIN 16 位
n2	强制为 OFF 的扫描个数	
Ⓓ	用户使用的定时时钟 (SM420 到 SM424, SM430 到 SM434)	位

[功能]

- (1) 为由Ⓓ指定的用户定时时钟 (SM420 到 SM424, SM430 到 SM434)，将 n1 个扫描周期变为 ON，并将 n2 个扫描周期变为 OFF。



- (2) 扫描执行类型程序使用 SM420 到 SM424，低速执行类型程序使用 SM430 到 SM434。
- (3) 如果 n1 和 n2 都被设置为 0，将会发生下列情况：
- (a) n=0 SM420 到 SM424 和 SM430 到 SM434 将保持 OFF。
 - (b) n>0, n2=0 SM420 到 SM424 和 SM430 到 SM434 将保持 ON。
- (4) 当 DUTY 指令被执行时，被 n1、n2 指定的数据和Ⓓ在系统中登记，并且定时脉冲被 END 处理变为 ON 和 OFF。

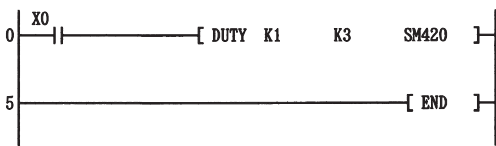
[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志（SM0）变为 ON，并且错误代码存储在 SD0 中。
- 被Ⓓ指定的软元件不是在从 SM420 到 SM424 或从 SM430 到 SM434 的范围内。(错误代码：4101)
 - n1 和 n2 的值小于 0。(错误代码：4100)

[程序示例]

(1) 如果 X0 为 ON，下列程序将 SM420 变为 ON 一个扫描周期，并将其变为 OFF 3 个扫描周期。

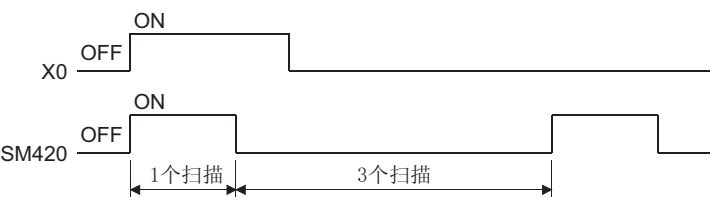
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	DUTY	K1 K3 SM420
5	END	

[运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*	○	○	○	○

＊1：系列号的前五位数是04122或更大的数。

7.18.3 时间检查指令 (TIMCHK)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\C[]		特殊功能模 块 U[]\C[]	变址寄存器 Zn
	位	字		位	字		
①	—	○					—
②					○		—
③	○						—

[指令符号]	[执行条件]
TIMCHK	

[设定数据]

设定数据	含义	数据类型
①	存储测量当前值的软元件	BIN 16 位
②	存储测量设定值的软元件。	
③	当时间到达时将会被变为 ON 的软元件	位

[功能]

- (1) 测量作为条件使用的软元件的 ON 时间，并且如果条件软元件保持 ON 的时间比由②指定的软元件中设定的时间要长，那么由③指定的软元件被变为 ON。
- (2) 由①指定的软元件的当前值被清除为 0，并且由③指定的软元件在输入条件的上升沿被变为 OFF。
如果输入条件变为 OFF，由①指定的软元件的当前值和由③指定的软元件的 ON 状态将被保持。
- (3) 测量的设定值是以 100ms 为单位设置的。

[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
 - 不能被指定的软元件已经被指定。(错误代码：4100)

[程序示例]

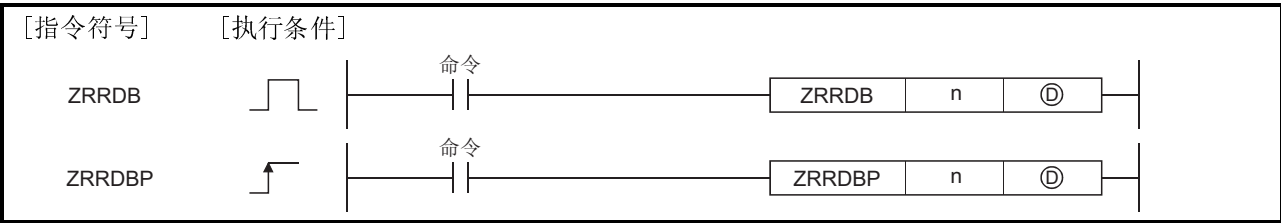
- (1) 在下面这个程序中，X0 的 ON 时间被设置为 5s，当前值存储软元件被设置为 D0，并且在时间到达时要变为 ON 的软元件被设置为 Y10。



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.18.4 从文件寄存器中的直接1字节读取 (ZRRDB, ZRRDBP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[] []		特殊功能模 块 U[] []	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
n	○						○	—	
⑤	○						—	—	

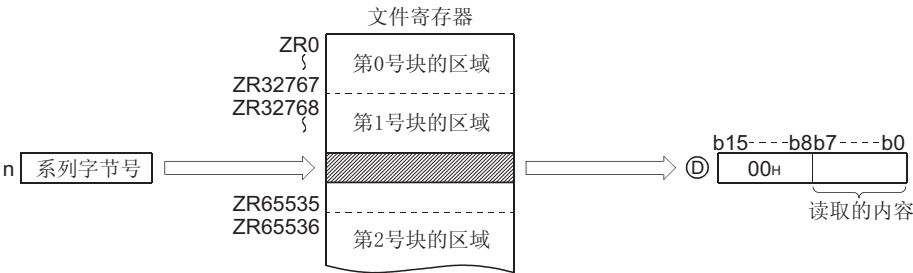


[设定数据]

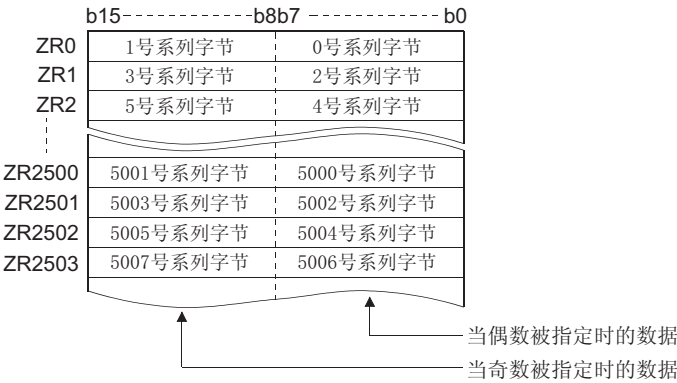
设定数据	含义	数据类型
n	将被读取的文件寄存器的系列字节号	BIN 32 位
ⓐ	存储读取数据的软元件的软元件号	BIN 16 位

[功能]

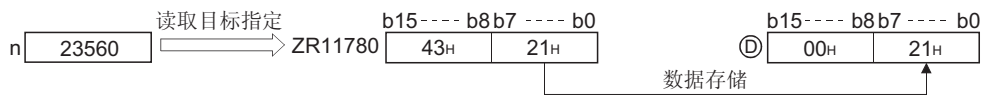
- (1) 读取被 n 指定的系列字节号，n 不表示块号，并且存储在由ⓐ指定的软元件中的低 8 位上。
由ⓐ指定的软元件的高 8 位将变为 00H。



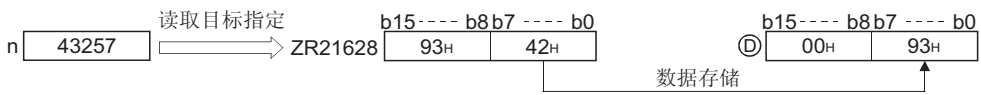
- (2) 文件寄存器号和系列字节号的对应关系如下：



(a) 如果 n 的值已被指定为 23560，将会读取在 ZR11780 中低 8 位上的数据。



(b) 如果 n 的值已被指定为 43257，将会读取在 ZR21628 中高 8 位上的数据。

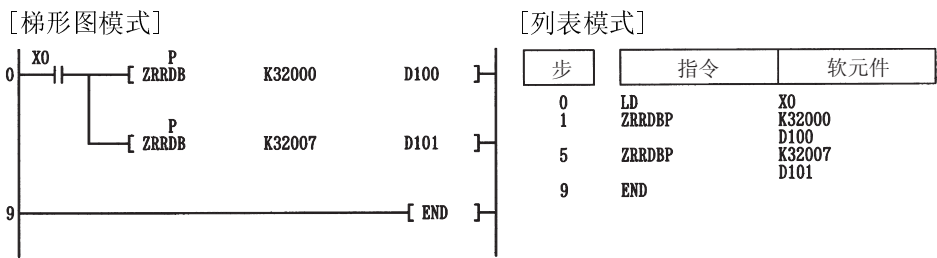


[运行错误]

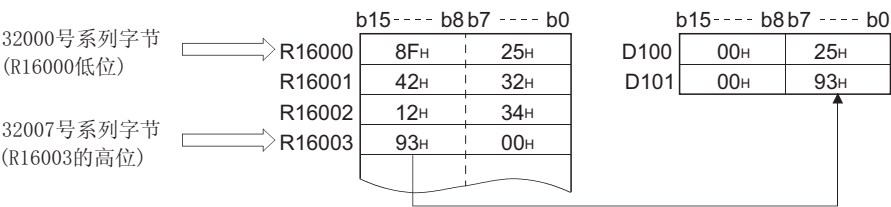
- (1) 在下列发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 超出允许的指定范围的软件件号（系列字节号）已经被指定。（错误代码：4101）

[程序示例]

- (1) 当 X0 为 ON 时，下列程序读取 ZR16000 的低位和 R16003 的高位，并且在 D100 和 D101 中存储结果。



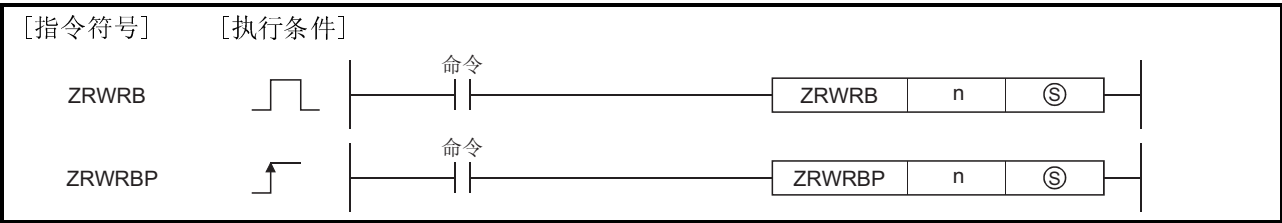
[运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.18.5 写入 1 字节到文件寄存器 (ZRWRB, ZRWRBP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]G[]		特殊功能模 块 U[]V[]G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
n	○								—
Ⓢ	○								—

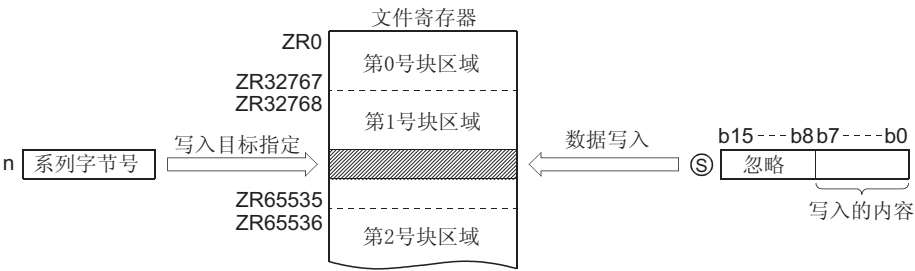


[设定数据]

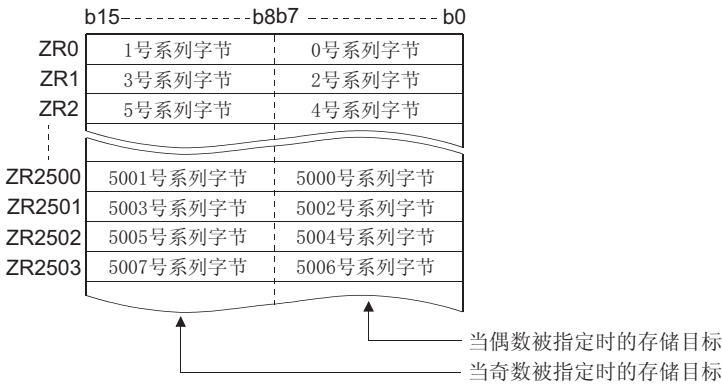
设定数据	含义	数据类型
n	要被写入的文件寄存器的系列字节号	BIN 32 位
Ⓢ	存储要被写入数据的软元件的软元件号	BIN 16 位

[功能]

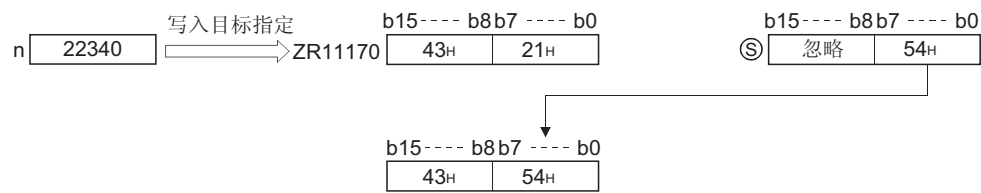
- (1) 将在由Ⓢ(不表示块号)指定的软元件中存储的数据写入到被 n 指定系列字节号的文件寄存器中去。
由Ⓢ指定的软元件中的数据的高 8 位将被忽略。



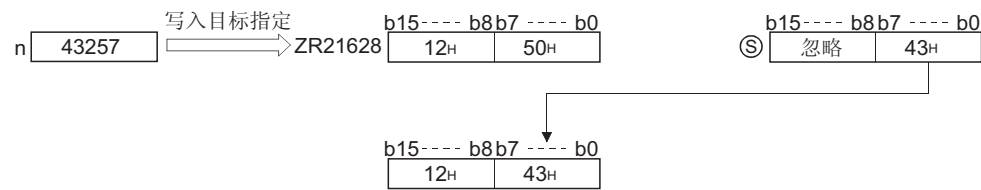
- (2) 文件寄存器号和系列字节号的对应关系如下：



如果 n 的值被指定为 22340，那么 ZR11170 的低 8 位将被写入。



当 n 值指定为 43257 时

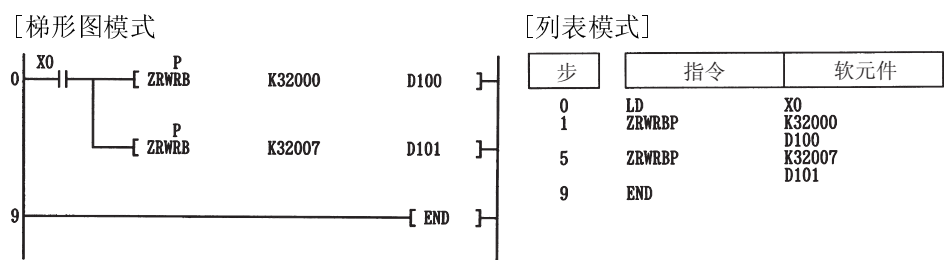


[运行错误]

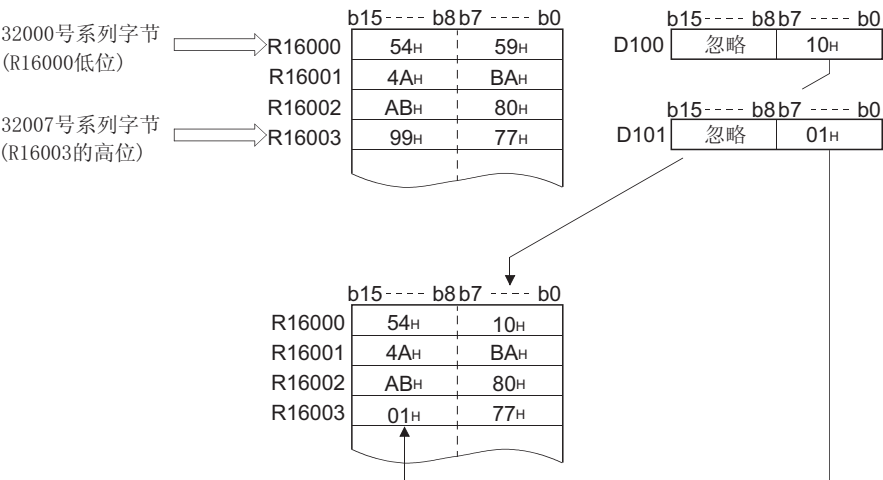
- (1) 在下列发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 超出允许的指定范围的软元件号（系列字节号）已经被指定。（错误代码：4101）

[程序示例]

- (1) 当 X0 变为 ON 时，下列程序将 D100 和 D101 中低位上的数据写入到 R16000 的低位和 R16003 的高位上去。



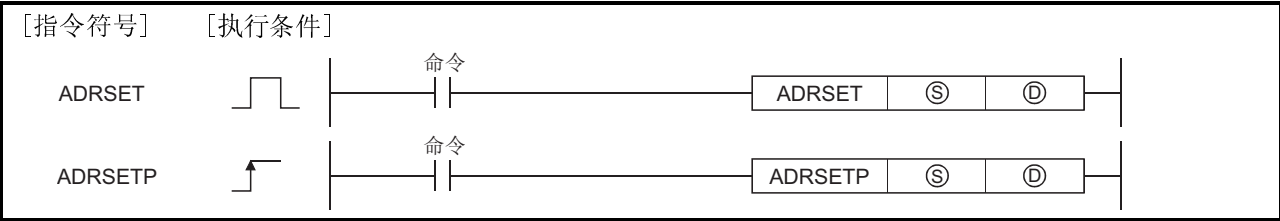
[运算]



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.18.6 间接地址读取操作 (ADRSET, ADRSETP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存 器 Zn
	位	字		位	字		常数
⑤	○			—			
⑥	○			—			

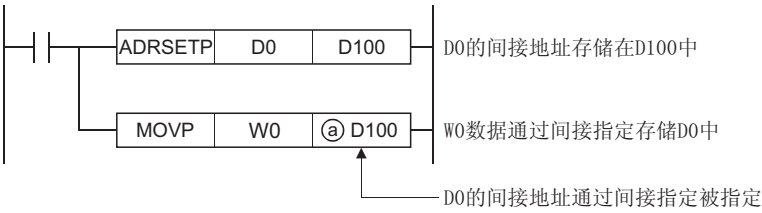


[设定数据]

设定数据	含义	数据类型
⑤	用于间接地址读取的软元件的软元件号	软元件名称
⑥	将存储由⑤指定的软元件的间接地址的软元件号	BIN 32 位

[功能]

- (1) 在⑥+1 和⑥中存储由⑤指定的软元件的间接地址。
当间接软元件地址被顺控程序执行时，使用由⑥指定的软元件中存储的地址。



- (2) 不能在⑤中进行位软元件指定。

[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 不允许指定的软元件已经被指定。(错误代码：4100)

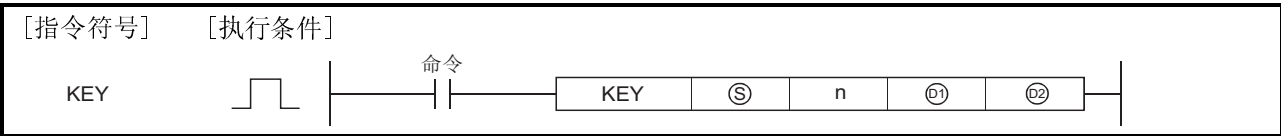
备注

*：关于间接指定的更多信息，请参见章节 3.4。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

7.18.7 键盘的数字键输入(KEY)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 [J][K][L]		特殊功能模 块 [U][V][G][L]	变址寄存器 Zn
	位	字		位	字		常数 K, H
⑤	○ (只 x)	—	—	—	—	—	—
n	○	○	—	—	○	—	—
①	—	○	—	—	—	—	—
②	○	○	—	—	○	—	—

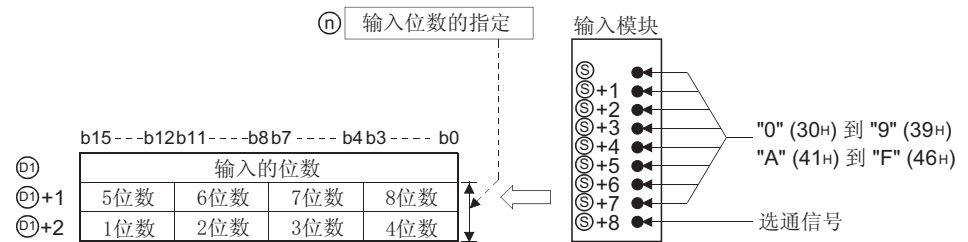


[设定数据]

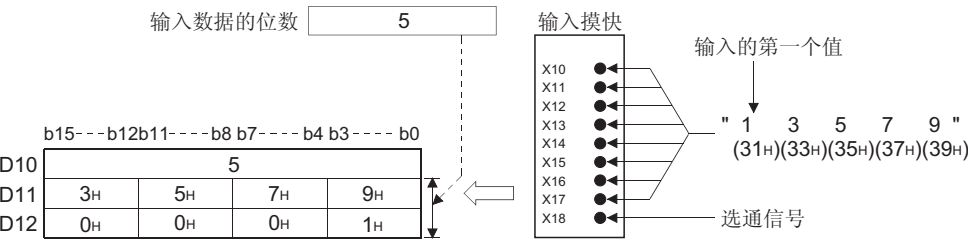
设定数据	含义	数据类型
⑤	要进行数字输入的软元件(X)的起始软元件号	位
n	将要进行的数字输入的位数	BIN 16 位
①	将存储数字输入的软元件的起始软元件号	
②	在输入结束时变为 ON 的位软元件的号码	位

[功能]

- (1) 从由⑤指定的输入(X)的 8 点中取出 ASCII 数据，将其转换为十六进制值，并且在从由①指定的软元件开始的软元件中存储结果。

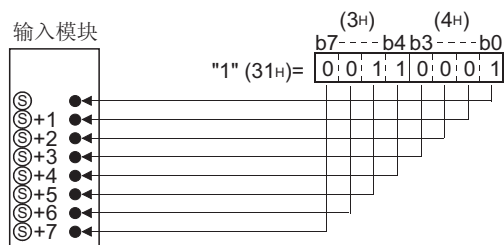


例如，当位数 (n) 已被设置为 5，并且已经通过输入模块的 X10 到 X18 输入值 “31”， “33”， “35”， “37” 和 “39” 时，将会发生下列情况：



- (2) 输入到由⑤指定的输入(X)的数字输入在⑤到⑤+7 上进行位扩展, 并按照相应的数字以 ASCII 码输入。

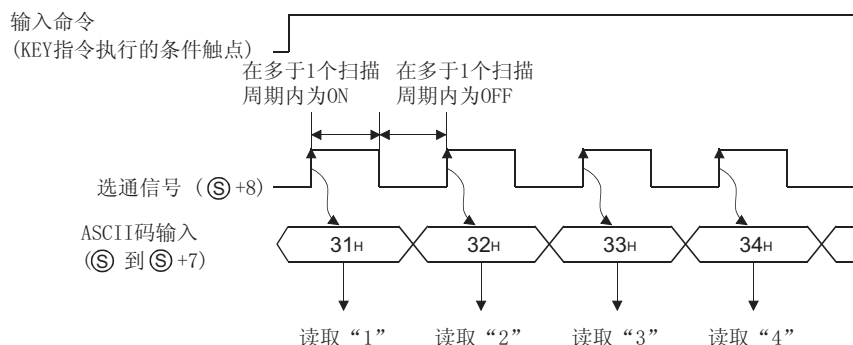
能够输入的 ASCII 码是在从 30_H(0)到 39_H(9)，和从 41_H(A)到 46_H(F)的范围之内的。



- (3) 在 ASCII 码输入到⑤到⑤+7 之后, 在⑤+8 上的选通信号变为 ON, 从而在内部合并指定的数字。

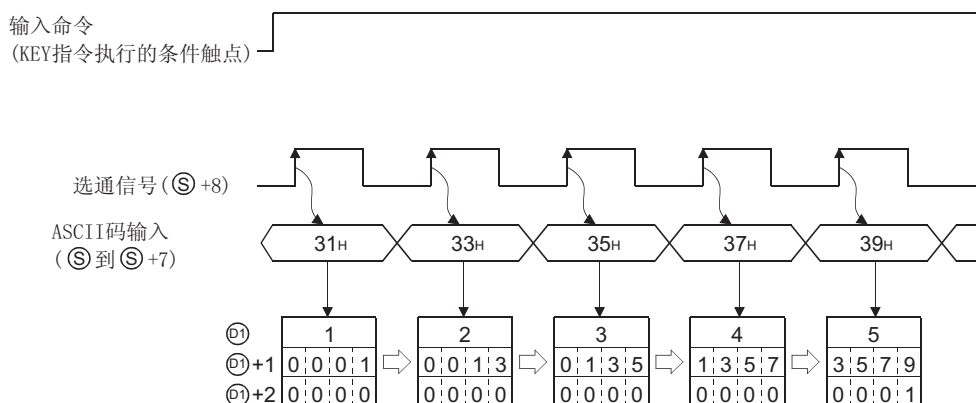
选通信号应该在顺控程序中的多于一个的扫描周期内保持其 ON 或 OFF 状态。

如果时间少于 1 个扫描周期, 当数据被正确合并时将会发生几种情况。



- (4) 在输入指定的位数之前，输入指令（KEY 指令执行条件触点）应该总保持 ON。如果输入指令变为 OFF，那么 KEY 指令就不能被执行。

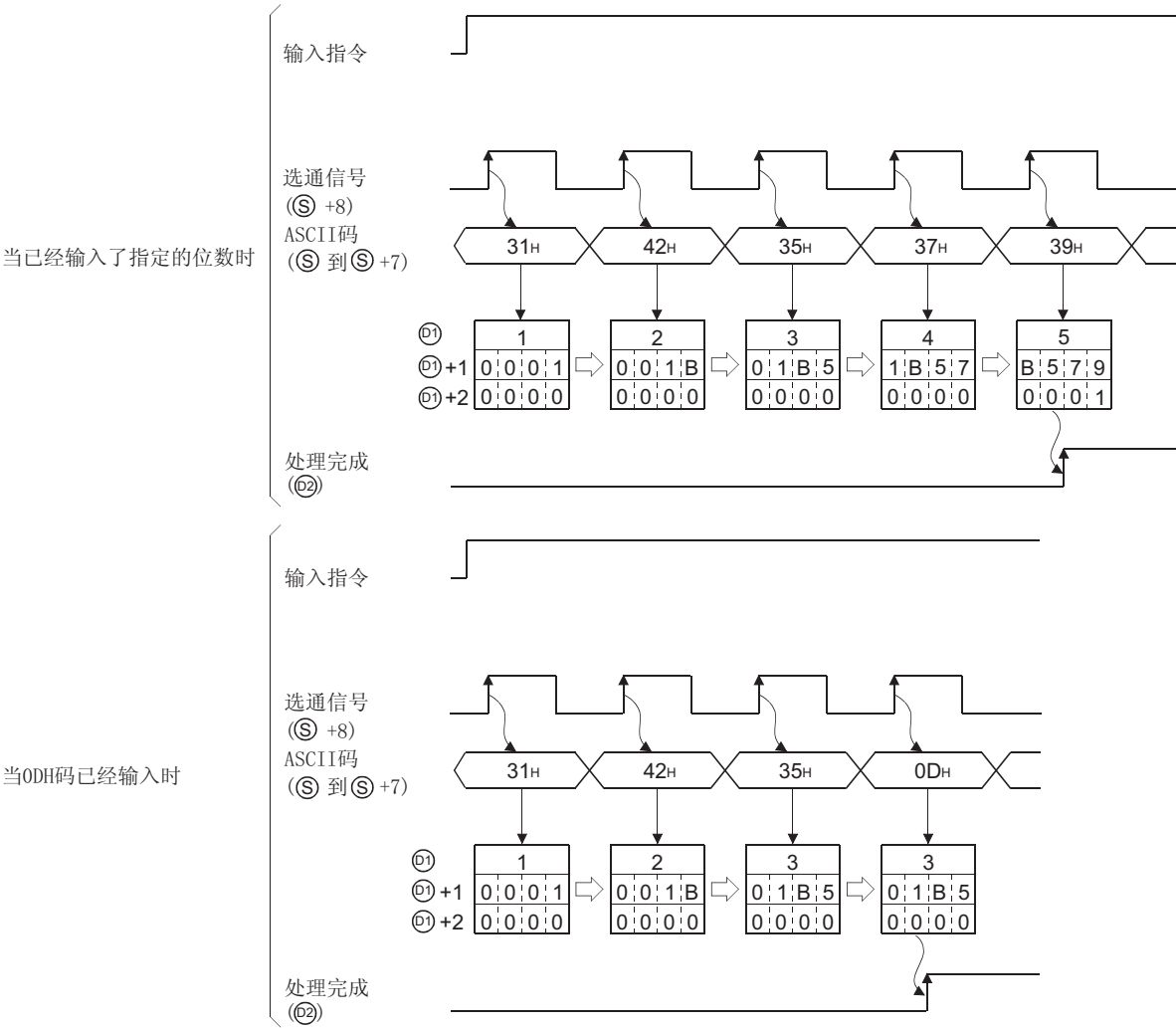
- (5) 实际从⑤中取出的数字的位数将会在⑥指定的软元件中存储，并且这些位数将会在⑥+1 和⑥+2 中转换为 ASCII 码，转换为十六进制 BIN 值，并且存储起来。



(6) 能够被(n)指定的位数为 1 到 8。

(7) 当已经输入由(n)指定的位数时，或当输入“0Dh”码时，输入数据的内部合并会结束，在这种情况下，由②指定的位软元件将会变为

例如，如果 n=5，在指定位置上的运算表示如下：



如果输入处理要执行第二次，就有必要清除存储在①上的输入数据和输入的位数，并且要运行一个用户程序，该用户程序将由②指定的位软元件变为 OFF。

如果①没有被清除并且②没有被变为 OFF，那下一个输入处理就不能被执行。

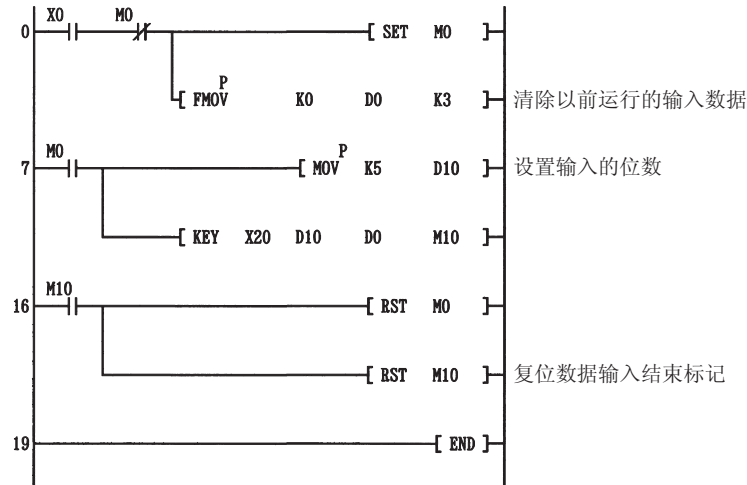
[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 被⑤指定的软元件不是一个输入(X)软元件。(错误代码: 4100)
 - 被 n 指定的位数在 1 到 8 的范围之外。(错误代码: 4100)

[程序示例]

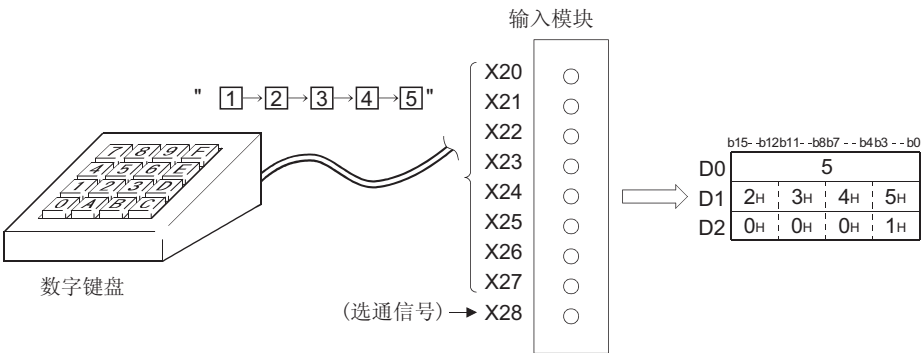
(1) 当 X0 变为 ON 时，下列程序合并来自连接的连接到 X20 到 X28 的数字键盘的 5 位或更少位的数据，并且在从 D0 开始的位置存储这些数据。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	ANI	M0
2	SET	M0
3	FMovP	K0 D0 K3
7	LD	M0
8	MOVP	K5 D10
11	KEY	X20 D10 D0 M10
16	LD	M10
17	RST	M0
18	RST	M10
19	END	



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	○	○

7.18.8 批量保存或变址寄存器的恢复 (ZPUSH, ZPUSHP, ZPOP, ZPOPP)

(1) 当使用基本类型 QCPU 时

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[][]		特殊功能模 块 U[][]G[][]	变址寄存器 Zn	常数	其它
	位	字		位	字				
①	—	○					—		

[指令符号]	[执行条件]	□ 表示 ZPUSH/ZPOP	
ZPUSH, ZPOP			
ZPUSHP, ZPOPP			

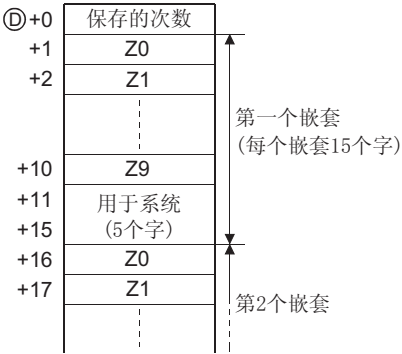
[设定数据]

设定数据	含义	数据类型
①	保存变址寄存器的软元件的起始软元件号	BIN 16 位

[功能]

ZPUSH

- (1) 将下列变址寄存器的内容保存到由①指定的软元件之后。
 - 基本类型 QCPU: Z0 到 Z9
- (2) ZPOP 指令是用于数据恢复的, 在 ZPUSH 到 ZPOP 的周期循环中可以使用嵌套。
- (3) 如果已经有嵌套, 每当 ZPUSH 指令被执行时, 在①之后使用的区域将会被添加上去, 所以这个区域从开始就要保持足够大, 从而能容纳该指令将要被使用的次数。
- (4) 在①之后使用的区域的组成如下:



ZPOP

(1) 保存到由Ⓔ指定的软元件之后的区域内的数据被读取到变址寄存器中。

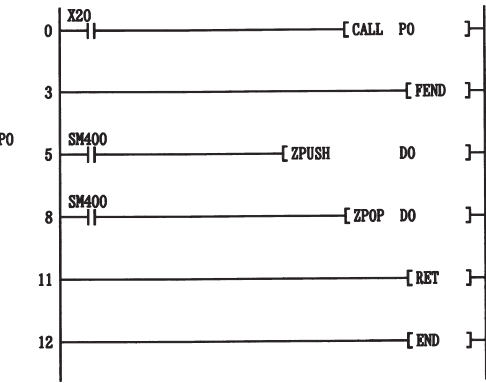
[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 要在Ⓔ中使用的和以后被 ZPUSH(P) 指令使用的点的个数范围超过了相应的软元件范围。(错误代码: 4101)
 - 在 ZPOP(P) 指令中Ⓔ+0 的内容(已进行保存的个数)为 0。(错误代码: 4100)

[程序示例]

(1) 下列程序在调用使用变址寄存器的 P0 之后的子函数之前，将变址寄存器的内容保存到 D0 之后的区域中去。

[梯形图模式]

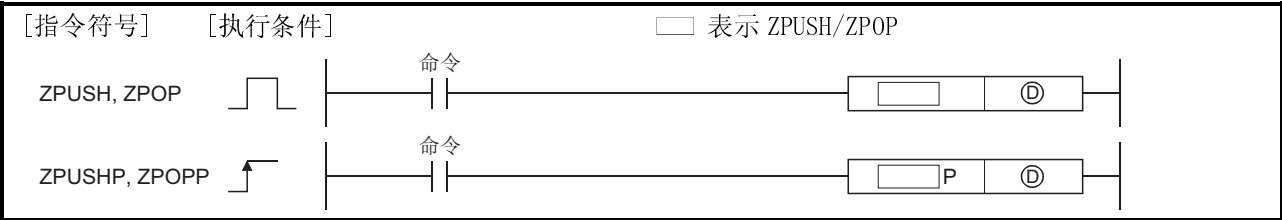


[列表模式]

步	指令	软元件
0	LD	X20
1	CALL	P0
3	FEND	
4		P0
5	LD	SM400
6	ZPUSH	D0
8	LD	SM400
9	ZPOP	D0
11	RET	
12	END	

(2) 当使用高性能型 QCPU/过程控制 CPU/QnACPU 时

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10(H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○					—



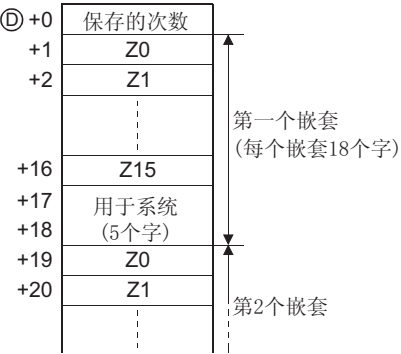
[设定数据]

设定数据	含义	数据类型
⑤	要保存变址寄存器的软元件的起始软元件号	BIN 16 位

[功能]

ZPUSH

- (1) 将下列变址寄存器的内容保存到由⑤指定的软元件之后。
- 高性能型 QCPU，过程控制 CPU，和 QnACPU: Z0 到 Z15
- (2) ZPOP 指令是用于数据恢复的指令，在 ZPUSH 到 ZPOP 周期循环中可以使用嵌套。
- (3) 如果已经有嵌套，每当 ZPUSH 指令被执行时，在⑤之后使用的区域将会被添加上去，所以这个区域从开始就要保持足够大，从而能容纳该指令将要被使用的次数。
- (4) 在⑤之后使用的区域的组成如下：



ZPOP

(1) 保存到由Ⓓ指定的软元件之后的区域内的数据被读取到变址寄存器中。

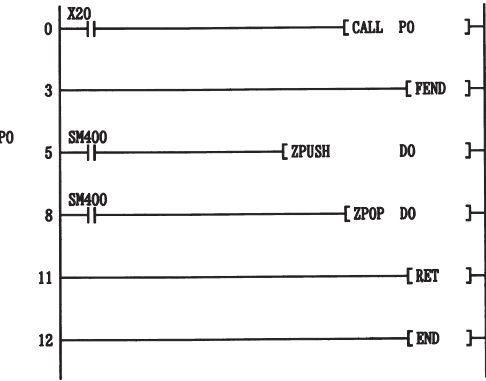
[运行错误]

- (1) 在下列发生运行错误的情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 要在Ⓓ中使用的和以后被 ZPUSH(P) 指令使用的点的个数范围超过了相应的软元件范围。
(错误代码: 4101)
 - 在 ZPOP(P) 指令中Ⓓ+0 的内容(已进行保存的个数)为 0。
(错误代码: 4100)

[程序示例]

(1) 下列程序在调用使用变址寄存器的 P0 之后的子函数之前，将变址寄存器的内容保存到 D0 之后的区域中去。

[梯形图模式]



[列表模式]

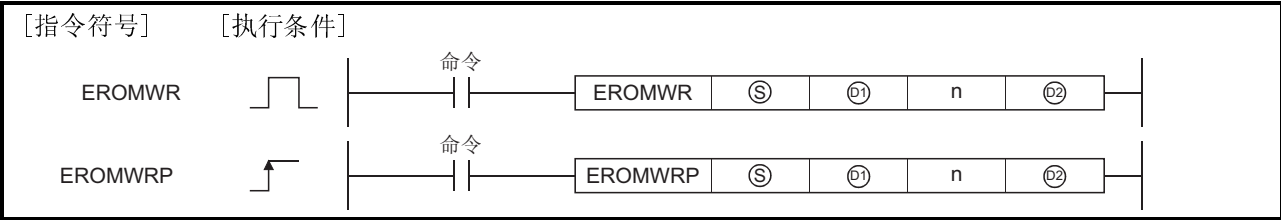
步	指令	软元件
0	LD	X20
1	CALL	P0
3	FEND	
4		P0
5	LD	SM400
6	ZPUSH	D0
8	LD	SM400
9	ZPOP	D0
11	RET	
12	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	×	×	○	○

7.18.9 对 EEPROM 文件寄存器的批写入操作 (EROMWR, EROMWRP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[]K[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				U
⑤	—	△*	—	—					—
⑥	—	—	△*	—					—
n	△*	△*	△*	○					—
⑦	△*	—	—	—					—

※:没有为本地软元件和程序文件设置的文件寄存器能被使用。



[设定数据]

设定数据	含义	数据类型
⑤	存储写入数据的软元件的起始软元件号	BIN 16 位
⑥	写目标文件寄存器的软元件的起始软元件号	
n	要写入的字数	
⑦	结束位软元件	位

[功能]

- (1) 从由⑤ 指定的软元件开始写入 n 个字的数据到由⑥指定的 E²PROM 驱动器中的文件寄存器中去。
- (2) 当写入操作已经结束时, 由⑦指定的位软元件被变为 ON, 并且在一个扫描周期之后再自动变为 OFF。
- (3) 数据写入操作是被 END 处理执行的, 在每一个处理周期内写入 64 个字。
因此, 指定字数的写入操作需要 n 除以 64 个扫描周期 (小数部分进位)。
另外注意, 处理过程的扫描时间被延长大约 10ms。
- (4) 不要在写入处理过程中更新⑤之后的数据。
如果在写入处理过程中更新了⑤之后的数据, 将会丢失一些数据。

[运行错误]

- (1) 在下列发生运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 被 n 指定的范围超过了⑤或⑥的相关软元件的范围。

(错误代码: 4101)
- 被⑥指定的文件寄存器不存在或者不是一个 E²PROM 文件寄存器。

(错误代码: 4101)

8. 数据链接指令

Q/QnACPU 允许使用 MELSECNET/H, MELSECNET/10, AJ71QE71, MELSECNET, CC-Link, 和 AJ71QC24 来进行数据链接。

数据链接指令是指允许 QnACPU 从连接到 MELSECNET/H, MELSECNET/10 或 MELSECNET 系统中的其它站中读取数据到主站中的指令。

(1) 数据链接指令类型

数据链接指令包括刷新指令, QnA 链接专用指令, 和 A 系列兼容链接指令。

- 刷新指令
：用于指定的网络模块刷新操作。
- QnA 链接专用指令
：具有 QnACPU 特征的数据链接新指令。
网络模块上从通道 1 到通道 8 的 8 个通道可用于执行通讯。
可以从控制数据区实现精确控制。
- A 系列兼容链接指令
：与 AnACPU/AnUCPU 专用指令相同的指令。

可以和 MELSECNET/H, MELSECNET/10 和 MELSECNET 一起使用的的数据链接指令是确定的。

并且, 对于 MELSECNET/H 和 MELSECNET/10, 根据在目标站中的 CPU 模块是 QCPU、 QnACPU、 ACPU 或远程 I/O 站, 指令的使用是不同的。

指令分类	访问源 CPU 模块	目标 CPU 模块		
		QCPU	QnACPU	ACPU
QnA 链接专用指令	QCPU *2	×	×	×
	QnACPU	○	○	×
	ACPU	×	×	×
A 兼容链接专用指令	QCPU *2	×	×	×
	QnACPU	○	○	○
	ACPU	×	×	×

表 8.1 数据链接指令

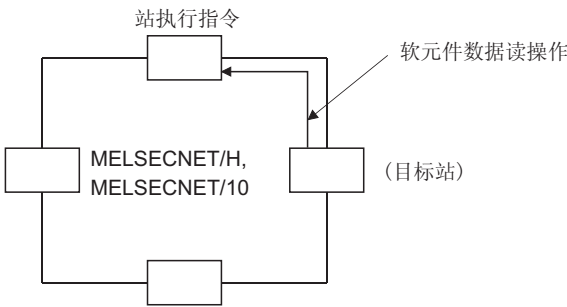
指令名		说明	参考章节
刷新指令	ZCOM	● 指定网络模块的刷新操作	8.1 节
QnA 链接专用指令	READ	● 从目标网络号的目标站中读取 QnACPU 软元件数据	8.2.1 节
	SREAD		8.2.2 节
	WRITE	● 将 QnACPU 软元件数据写入目标网络号的目标站	8.2.3 节
	SWRITE		8.2.4 节
	SEND	● 将数据写入目标网络号的目标站 (网络模块)	8.2.5 节
	RECV	● 在 QnACPU 读取由 SEND 指令发送的数据	8.2.6 节
	REQ	● 读/写时钟数据到其它站并传输远程 RUN/STOP	8.2.7 节
	ZNFR	● 从远程 I/O 站的特殊功能模块读取数据	8.2.8 节
	ZNT0	● 将数据写入远程 I/O 站的特殊功能模块	8.2.9 节
A 系列兼容链接指令	ZNRD	● 从目标网络号的目标站读取 QnACPU 软元件数据	8.3.1 节
		● 读就地站的软元件数据 (只有主站有效)	8.3.2 节
	ZNWR	● 将 QnACPU 软元件数据写入目标网络号的目标站	8.3.3 节
		● 将软元件数据写入就地站 (只在主站有效)	8.3.4 节
	RFRP	● 从远程 I/O 站的特殊模块读取数据	8.3.5 节
	RTOP	● 将数据写入远程 I/O 站特殊功能模块	8.3.6 节
路由信息, 读/写指令	RTREAD	● 读取路由参数指定的数据	8.4.1 节
	RTWRITE	● 将路由数据写入路由参数指定的区域	8.4.2 节

备注

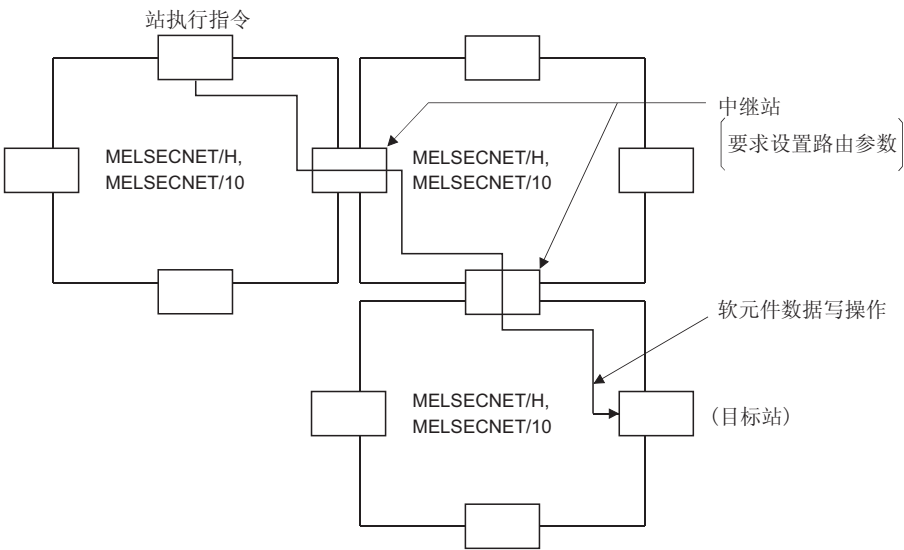
- *1: MELSECNET/H、MELSECNET/10 和 MELSECNET 解释如下：
- MELSECNET/H..... MELSECNET/H 网络系统
MELSECNET/10..... MELSECNET/10 网络系统
MELSECNET..... MELSECNET (II, /B) 数据链接系统
- *2: QCPU 可用的指令在相应的链接模块说明手册中作了解释。
可用指令的详细内容参照下列说明手册。
- Q 相应的 MELSECNET/H 网络系统参考手册 (PLC PLC 网络)。
 - Q 相应的以太网接口模块用户手册 (应用)。

(2) 数据读/写范围

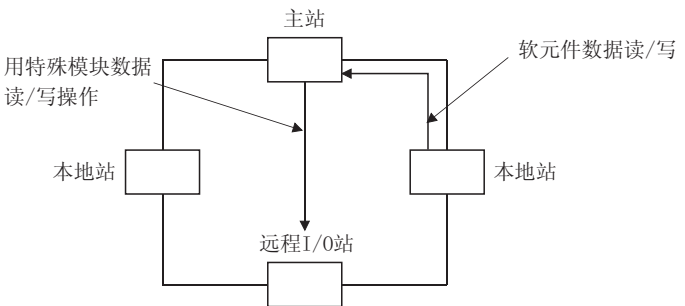
- (a) 使用 MELSECNET/H 和 MELSECNET/10 指令，一个主站不仅可以连接到它自己网络上的站，而且可以使用连接到指定网络号的站执行读/写操作。
- ① 在同一网络里对站进行读/写操作
- 使目标站的网络号与连接到自站的网络模块的网络号一致。
此功能是用从自站读写软元件数据到目标网络号上的任何其它站。



- ② 连接到指定网络号上的站的读/写操作
- 目标站上的网络号不同于自站网络模块使用的网络号。
与自站连接到同一网络上的站是用作在连接到指定网络号上的站上对软元件中转读/写操作的中继站。



(b) 对远程站或本地站的读写操作可以从主站用 MELSECNET (II/B) 系统进行。



要点	
可以由数据链接指令读/写数据的其它站 CPU 模块的软元件必须在自站 CPU 模块的应用软元件范围内指定。	
当其它站 CPU 模块与自站 CPU 模块应用的软元件范围不同时必须注意。	

(3) QnA 链接专用指令

下列是使用专用 QnA 链接指令时的注意事项。

- (a) 数据链接 MELSECNET/10 网络模块的同步执行指令有八个使用数据链接指令的通讯区域。
用相应的一个通讯区域，同时用 MELSECNET/10 网络模块对数据链执行多指令是不可能的。
如果在 QnACPU 同样的通讯区域用来执行数据链接的一个以上的指令，则由程序来完成，此程序将用数据链接单个指令的结束软元件逐步执行。

(b) 传送结束确认

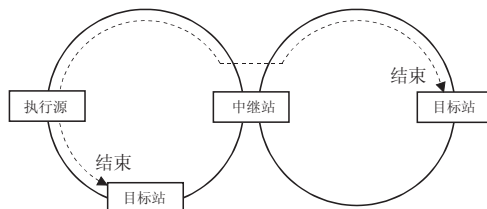
当用 QnA 专用链接指令进行数据传送时，可以选择传送结束确认或不确认。

(当用 QnA 专用链接指令进行读操作时，只能选择结束确认)

● 传送结束确认

：此为对指定目标站指定通道数据写入的指令结束。

因为处理过程不等待对目标站的写结束通知，因此执行源的指令结束就会更早。

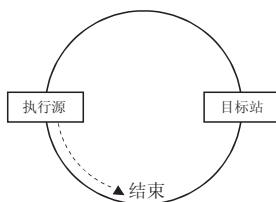


● 无传送结束确认

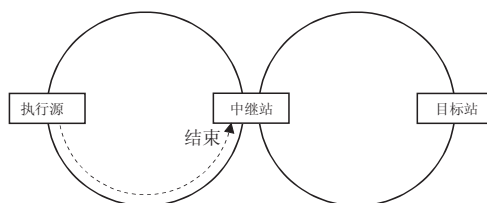
：目标站与自站在同一网络上时，从自站传送了数据时指令结束。

如果目标站在另外一个网络上，当数据到达主网络上中继站时即为指令结束。

[当目标站在自站网络中时]



[当目标站在不同网络中时]



要点
(1) 为了提高数据可靠性, 建议设定“传送结束确认”并执行此指令。 (2) 如果设定成“无传送结束确认”, 如果传送的数据有错误时, 只要通讯本身正常结束, 传送站将正常结束。 当从多站执行指令到相应同一站时, 即使传送的数据内容是正确的, 目标站仍然发送“接受缓冲器满”错误。 然而, 在这种情况下, 传送站将送出正常结束。

- (4) 网络模块通道
网络模块有八个通道来执行指令, 八个通道可以同时使用, 但同一通道不能被多条指令使用。
建立一个程序来执行与特殊继电器 (SB) 的互锁, 使多指令不能执行。



每一指令的互锁信号说明如下：

指令	ZNRD * 1	ZNWR * 2	—	—	—	—	—	—
	SEND, RECV, READ, WRITE, 和 REQ 指令							
	通道 1	通道 2	通道 3	通道 4	通道 5	通道 6	通道 7	通道 8
第 1	SB030	SB032	SB034	SB036	SB038	SB03A	SB03C	SB03E
第 2	SB230	SB232	SB234	SB236	SB238	SB23A	SB23C	SB23E
第 3	SB430	SB432	SB434	SB436	SB438	SB43A	SB43C	SB43E
第 4	SB630	SB632	SB634	SB636	SB638	SB63A	SB63C	SB63E

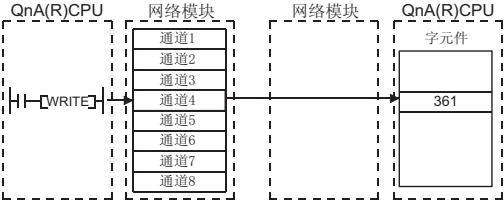
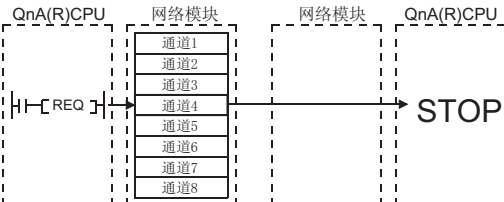
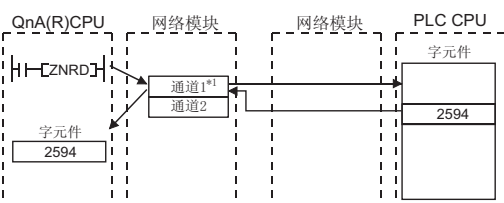
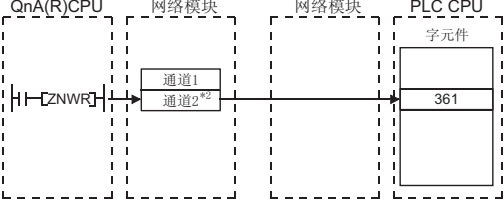
* 1 ZNRD 总是使用通道 1。
* 2 ZNWR 总是使用通道 2。

此部分描述了可以使用 MELSECNET/H, MELSECNET/10 的指令，每一条指令概述如下。

指令	详细说明	指令执行站(自站)	目标站	
		站类型	站类型	PLC CPU 型号
				QnA (R) CPU QnA (R) CPU 以外其它型号
发送接收	QnA (R) CPU 站之间数据发送 (SEND) 与接收 (RECV) 	控制站 普通站 远程主站 多远程主站 并行远程主站 多远程子主站 并行远程子主站	控制站 普通站 远程主站 多远程主站 并行远程主站 多远程子主站 并行远程子主站	○ ×
READ SREAD	从其它站的字软元件中读数据 (通过 SREAD, 可以打开目标站上的软元件) 	控制站 普通站 远程主站 多远程主站 并行远程主站 多远程子主站 并行远程子主站	控制站 普通站 远程主站 多远程主站 并行远程主站 多远程子主站 并行远程子主站	○ ×

* 通道 1 到 8 是 SEND/RECV/READ/WRITE/REQ 指令的公共区域。

在指令格式上没有操作区别 JP. 和 GP. , 和 J. 和 G. .

指令	详细内容	指令执行站(本站)	目标站			
		站类型	站类型	PLC CPU 型号		QnA (R) CPU 以外 其它型号
				QnA (R) CPU	QnA (R) CPU 以外 其它型号	
WRITE SWRITE	<p>将数据写到另外一个站的字软元件(用 SWRITE 指令, 可以打开目标站上的软元件)</p> 	控制站 普通站 远程主站 多远程主站 并行远程主站 多远程子主站 并行远程子主站	控制站 普通站 远程主站 多远程主站 并行远程主站 多远程子主站 并行远程子主站	○	×	
REQ	<p>对其它站执行”远程 RUN/STOP” “时钟数据读和写”</p> 	控制站 普通站 远程主站 多远程主站 并行远程主站 多远程子主站 并行远程子主站	控制站 普通站 远程主站 多远程主站 并行远程主站 多远程子主站 并行远程子主站	○	×	
ZNRD	<p>从另一个站的字软元件中读数据</p>  <p>* 1: 通道 1 固定</p>	控制站 普通站 远程主站 多远程主站 并行远程主站 多远程子主站 并行远程子主站	控制站 普通站 远程主站 多远程主站 并行远程主站 多远程子主站 并行远程子主站	○	○	
ZNWR	<p>对另一个站的字软元件写数据</p>  <p>* 2: 通道 2 固定</p>	控制站 普通站 远程主站 多远程主站 并行远程主站 多远程子主站 并行远程子主站	控制站 普通站 远程主站 多远程主站 并行远程主站 多远程子主站 并行远程子主站	○	○	

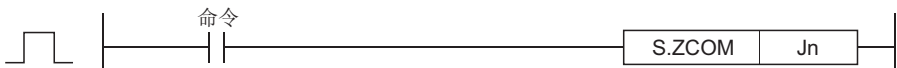
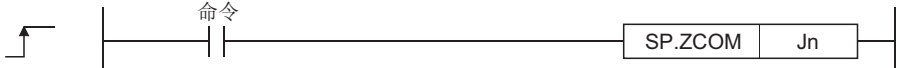
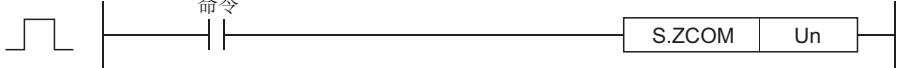
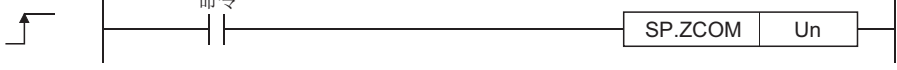
QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能性			
○	○	○	○	○

8.1 网络刷新指令

8.1.1 网络刷新 (ZCOM)

(1) 使用了 QCPU 时

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模块 U[][]	变址寄存器 Zn	常数	其它
	位	字		位	字				
	—								

[指令符号]	[执行条件]
ZCOM	
ZCOM	
ZCOM	
ZCOM	

[设定数据]

设定数据	说明	数据类型
Jn	自站网络号	BIN 16 位
Un	自站网络起始 I/O 号	

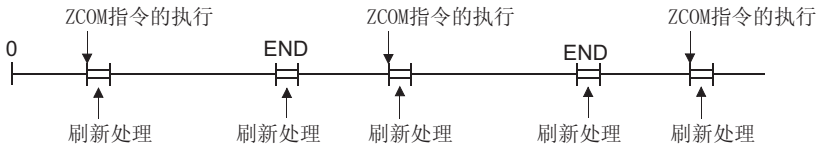
(1) ZCOM 指令是在顺控程序执行的任何时间执行刷新。

ZCOM 指令执行的刷新目标说明如下：

- MELSECNET/H 刷新(当设定了刷新参数时)
- CC-Link 的自动刷新(当设定了刷新软元件时)
- 智能功能模块的自动刷新(当设定了自动刷新时)

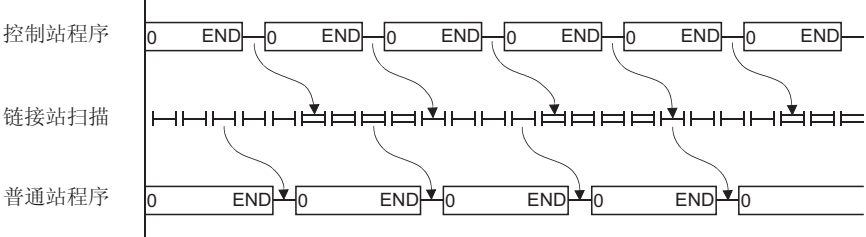
[功能]

(1) 当执行了 ZCOM 指令后, CPU 模块暂时停止处理顺控程序, 而执行由 Jn/Un 指定的网络模块的刷新处理。

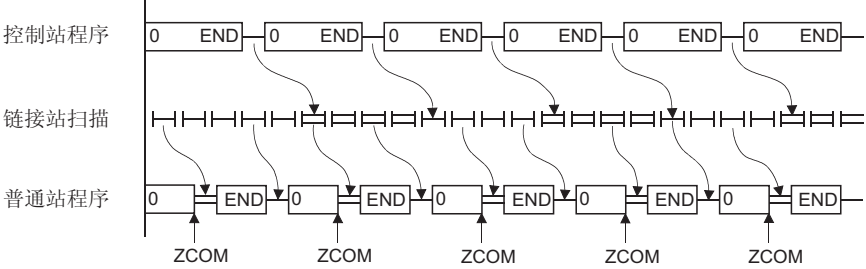


- (2) ZCOM 指令不执行下列通用数据处理
 - (a) CPU 模块与编程工具之间的通讯处理
 - (b) 其它站的监视处理
 - (c) 通过串行通讯模块，读取另外一个智能功能模块缓冲内存的处理
 - (d) MELSECNET/H 数据的低速周期传送
- (3) PLC 到 PLC 网络
 - (a) 当自站顺控程序的扫描时间长于其它站的扫描时间时，ZCOM 指令用来保证来自其它站的数据被正确合并。

① 不用 ZCOM 指令的数据通讯举例

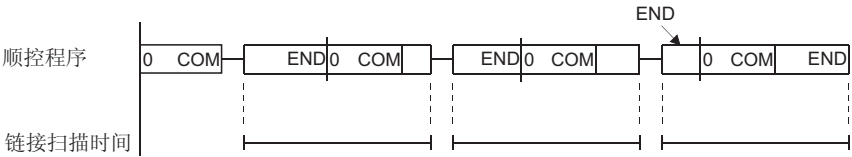


② 使用 ZCOM 指令的数据通讯举例



从 PLC 到 PLC 的网络传送延迟时间的详细说明，请参照 Q 对应的 MELSECNET/H 网络系统参考手册（PLC 到 PLC 网络）。

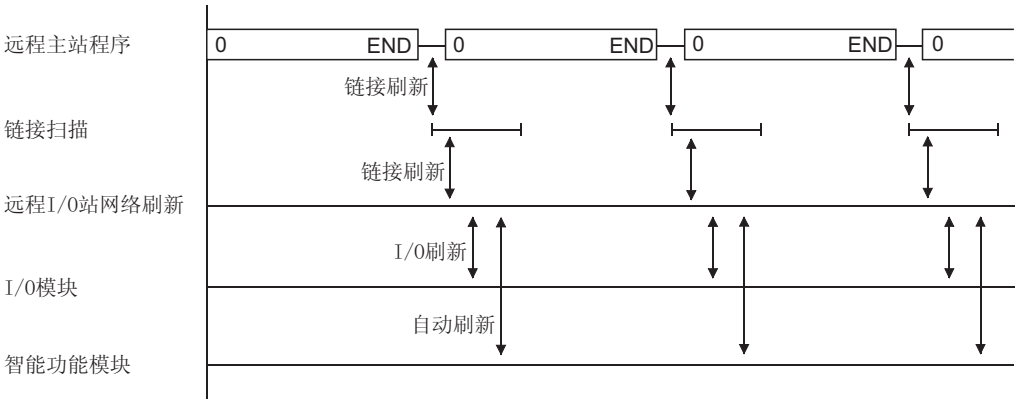
- (b) 当链接扫描时间长于顺控程序扫描时间时，即使使用 ZCOM 指令数据通讯也不能进行。



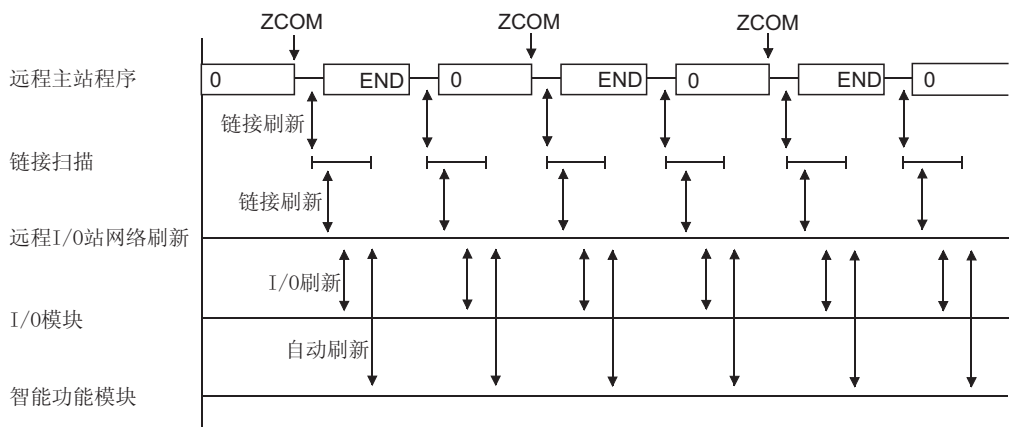
(4) 远程 I/O 网络

远程主站的链接刷新由 CPU 模块的“END 处理”来执行。
因为链接扫描是在链接刷新结束时执行, 链接扫描与 CPU 模块程序“同步”。
在远程主站使用“ZCOM”指令时, 链接刷新在 ZCOM 指令执行点执行, 并且在链接刷新结束时执行链接扫描。
因此, 在远程主站使用 ZCOM 指令可以加速远程 I/O 站的发送或接收处理速度。

① 不使用 ZCOM 指令时



② 使用 ZCOM 指令时



远程 I/O 网络的传送延迟时间的详细内容, 请参照 Q 对应的 MELSECNET/H 网络系统参考手册 (远程 I/O 网络)。

(5) ZCOM 指令可以在顺控程序中无限量使用, 然而, 每执行一次刷新操作都将使顺控程序扫描时间延长, 并增加了刷新操作所需要的时间。

- (6) 在自变量中指定“Un”可将目标指定为网络模块或智能模块。
在这种情况下，智能功能模块的缓冲内存将自动刷新(代替 FROM/T0 指令)。

要点	
ZCOM 指令不能用于固定周期执行类型的程序或中断程序。	

[运行错误]

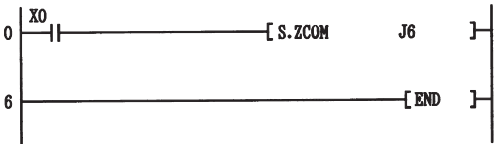
- (1) 在以下发生运行错误情况中，错误指示(SM0)变为 ON，并且错误代码存储在 SD0 中。
- 指定的网络号没有连接到本站(错误代码：4102)。
 - 指定首 I/O 号的模块不是网络模块或链接模块(错误代码：2111)。

要点	
只进行公共数据处理, 使用 COM 指令(见 7.6.8 节)。	

[程序示例]

- (1) 当 X0 变为 ON 时，下列程序对网络号为 6 的网络模块进行链接刷新操作。

[梯形图模式]

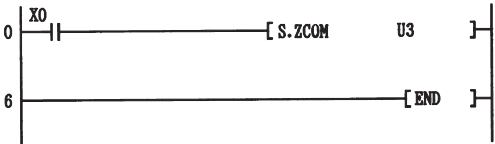


[列表模式]

步	指令	软元件
0	LD	X0
1	S.ZCOM	J6
6	END	

- (2) 当 X0 变为 ON 时，下列程序对安装在 I/O 号从 X/Y30 到 X/Y4F 的网络模块进行链接刷新操作。

[梯形图模式]

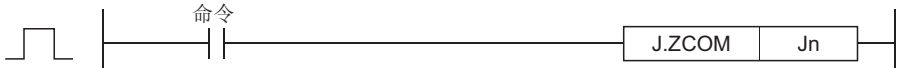
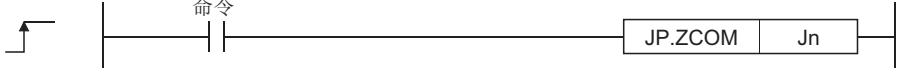
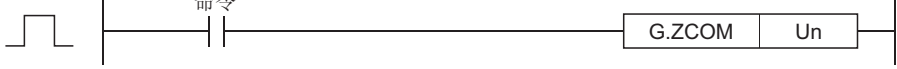
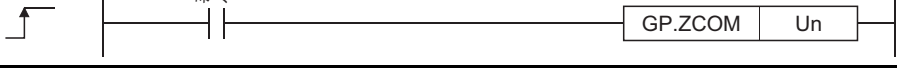


[列表模式]

步	指令	软元件
0	LD	X0
1	S.ZCOM	U3
6	END	

(2) 使用 QnACPU 时

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模块 U[]\GE[]	变址寄存器 Zn	常数	其它
	位	字		位	字				
—	—								

[指令符号]	[执行条件]
ZCOM	
ZCOM	
ZCOM	
ZCOM	

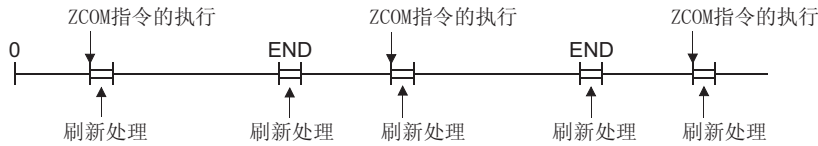
[设定数据]

设定数据	说明	数据类型
Jn	自站网络号	BIN 16 位
Un	自站网络起始 I/O 号	

- (1) 在下列情况下使用 ZCOM 指令。
- (a) 加速与远程 I/O 站的通讯处理速度
 - (b) 在数据链接执行过程中, 需要与不同扫描时间的其它站进行稳定的数据交换

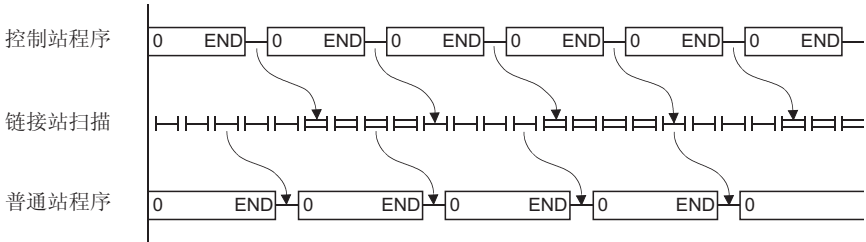
[功能]

- (1) 当执行了 ZCOM 指令时, QnACPU 暂时停止顺控程序的处理, 来进行 Jn/Un 指定的网络模块的刷新处理。
- 注意不能执行 MELSECNET/10 的慢速周期刷新。

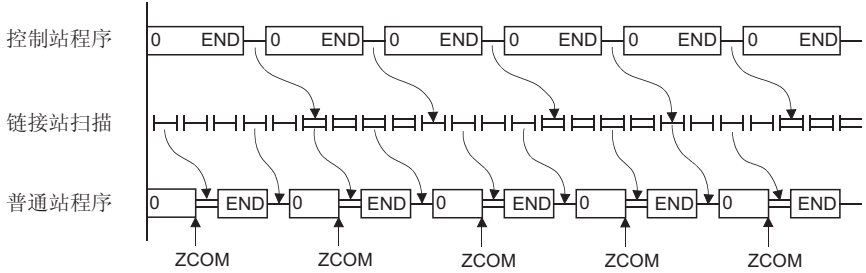


- (2) ZCOM 不能执行下列一般数据处理。
 - (a) CPU 模块与编程工具间的通讯处理
 - (b) 其它站的监视处理
 - (c) 通过串行通讯模块对另一个智能功能模块缓冲内存的读处理
- (3) PLC 到 PLC 网络
 - (a) 当自站顺控程序的扫描时间长于其它站的扫描时间，ZCOM 指令用来保证来自其它站的数据正确组合。

① 当没有使用 ZCOM 指令时，数据通讯举例

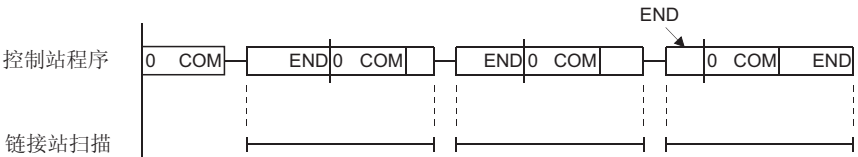


② 当使用了 ZCOM 指令时，数据通讯举例



PLC 到 PLC 网络的传送延迟时间的详细内容，参考 QnA/Q4AR MELSECNET/10 网络系统参考手册。

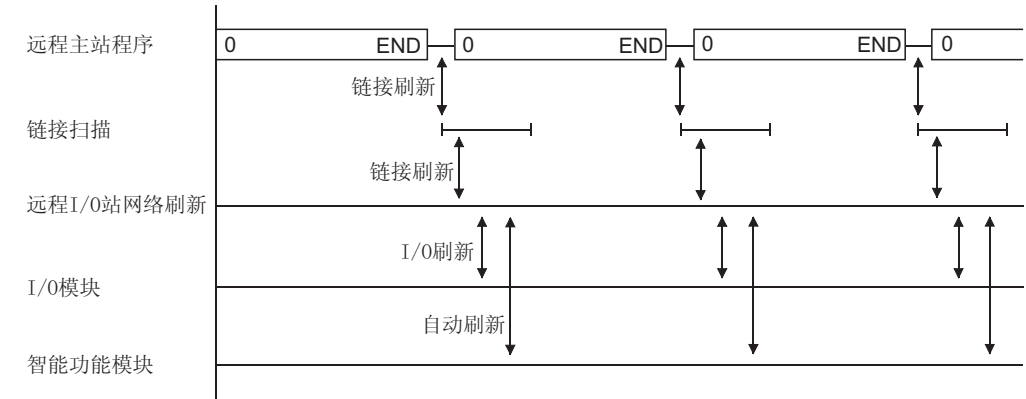
- (b) 当链接扫描时间长于顺控程序扫描时间，即使使用 ZCOM 指令也不能进行数据通讯。



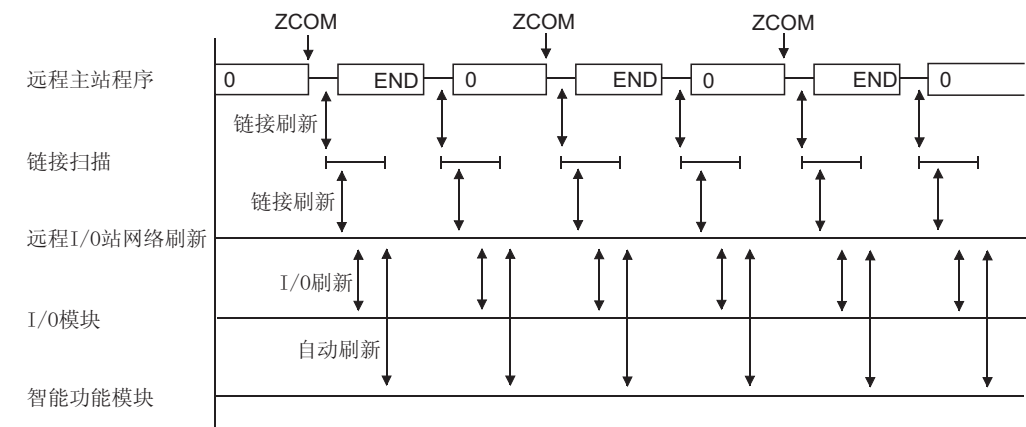
(4) 远程 I/O 网络

通过 CPU 模块的“END 处理”执行远程主站的链接刷新。
因为链接扫描是在链接刷新结束时执行, 链接扫描与 CPU 模块程序“同步”。
在远程主站使用 ZCOM 指令时, 链接刷新是在 ZCOM 指令执行点完成的, 而链接扫描是在链接刷新结束完成的。
因此, 在远程主站使用 ZCOM 指令可加速远程 I/O 站的发送/接收处理速度。

① 没有使用 ZCOM 指令时



② 使用了 ZCOM 指令时



远程 I/O 网络传送延迟时间的详细内容, 请参考 Q 对应的 MELSECNET/H 网络系统参考手册 (远程 I/O 网络)。

(5) ZCOM 指令可以在顺控程序中无限次使用。

然而, 每进行一次刷新操作都将使顺控程序扫描时间延长, 增加了刷新操作所需要的时间。

[运行错误]

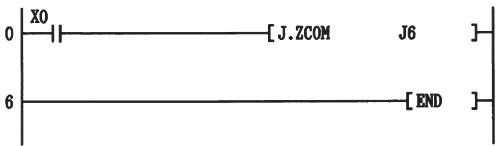
- (1) 在以下发生运行错误情况中。错误指示 (SM0) 开，并且错误代码储存在 SD0 中。
- 指定的网络号没有连接到自站上 (错误代码:4102)
 - 指定首 I/O 号的模块不是网络模块或链接模块 (错误代码:2111)

要点
只进行公共数据处理，使用 COM 指令 (见 7.6.8 节)

[程序示例]

- (1) 当 X0 变为 ON 时，下列程序将对网络号为 6 的网络模块进行链接刷新操作。

[梯形图模式]

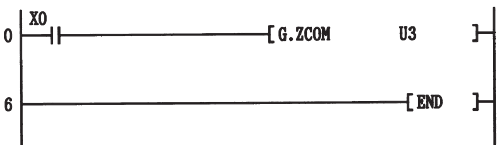


[列表模式]

步	指令	软元件
0	LD	X0
1	J.ZCOM	J6
6	END	

- (2) 当 X0 变为 ON 时，下列程序将对安装在 I/O 号为从 X/Y30 到 X/Y4F 的网络模块进行链接刷新操作。

[梯形图模式]



[列表模式]

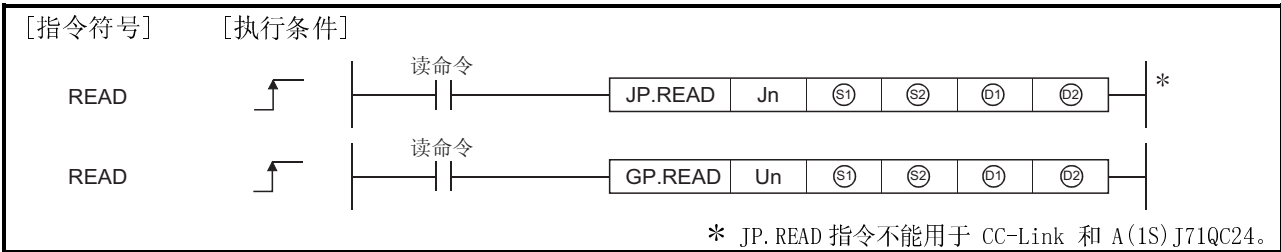
步	指令	软元件
0	LD	X0
1	G.ZCOM	U3
6	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×*4	×*4	×*4	○	○

8.2 QnA 链接专用指令

8.2.1 从另一个站中读字软元件数据 (READ)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[]\G[]		特殊功能模块 U[]\G[]	变址寄存器 Zn	常数	其它
	位	字		位	字				
①	—	○	—						
②	—	○	—						
③	—	○	—						
④	○	○	—						



[设定数据]

MELSECNET/10, A(1S)J71QE71

设定数据	说明	数据类型
Jn	自站网络号 *1	BIN 16 位
Un	网络模块自站的起始 I/O 号 *2	
①	自站存储控制数据的起始软元件号	软元件名称
②	读取数据所储存站的起始软元件号	
③	自站存储读数据的起始软元件号	
④	指令结束时, 软元件将在 1 次扫描期间为 ON。 *3	位

备注

- *1 : 自站网络号按如下指定
 - 1 到 239 : 网络号
 - 254 : 由设定为其它站可以访问的有效模块来指定的网络
- *2 : 自站网络模块的起始 I/O 号的指定范围为 0 到 FEn。
- *3 : 本地软元件和每个程序的文件寄存器不能用作数据设定软元件 (详细内容参考 QnACPU 编程手册 (特殊功能模块)。)
- *4 : QCPU 可以使用的 READ 指令的详细内容, 请参考以下手册。
 - Q 相应的 MELSECNET/H 网络系统参考手册 (PLC 到 PLC 网络)。
 - Q 相应的以太网接口模块用户手册 (应用篇)。

要点

- (1) 只有当目标站是 QCPU/QnACPU 时 READ 指令才能被执行。
(READ 指令不能在连接到 MELSECNET/10 的 ACPU 中执行。)
只有 QCPU/QnACPU 的站号应该设定成目标站号。
- (2) 当用 READ 指令从其它站 CPU 模块软元件读数据时, 在自站 CPU 模块内指定软元件。
(其它站 CPU 模块要读的起始软元件号②) + (读的点数 - 1) ≤ (自站 CPU 模块的最后软元件号 *)
*: 自站 CPU 模块里的最后软元件号与②有同样的软元件名。

关于 CC-Link A(1S)J71QC24

CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

设定数据	说明	数据类型
Un	主/本模块自站的起始 I/O 号 *1	BIN 16 位
Ⓔ1	本站控制数据的起始软元件	软元件名
Ⓔ2	站中存储了读取数据的起始软元件	
Ⓔ1	本站存储读取数据的起始软元件	
Ⓔ2	指令结束后, 在一个扫描周期内软元件为 ON。 *2	位

备注

*1 自站的主/自站模块起始 I/O 号的指定范围为从 0 到 FEH。

*2 本地软元件和每个程序的寄存器不能用于数据设定软元件。

(详细信息见 QnACPU 编程手册(特殊功能模块))

[控制数据]

用于 MELSECNET/10, A(1S)J71QE71

软元件	条目	设定数据	设定范围	设定者
Ⓔ+0	执行类型	到达确认(位 0=1; 固定方式)	0001h 0081h	用户
	错误结束类型	当错误处理完成后设定时钟数据设定状态 不设定时钟数据 : 将位 7 (b7) 设定成 0 设定时钟数据 : 将位 7 (b7) 设定成 1 (Ⓔ+11 在前面添加)		
Ⓔ+1	结束状态	指令结束被存储后的状态 0 : 没有错误 (正常结束) 不是 0 : 错误代码 *1	—	系统
Ⓔ+2	自站使用的通道	指定自站使用的通道 *2	1 到 8	用户
Ⓔ+3	虚拟	未使用	0	—
Ⓔ+4	目标站网络号	设定指定给读取软元件数据站点的网络号	1 到 239 254 *3	用户
Ⓔ+5	目标站号	设定指定站点的站点号	1 到 64	用户
Ⓔ+6	虚拟	未使用	—	—
Ⓔ+7	传送重试次数	如果 READ 指令提示的读取操作在Ⓔ+8 指定的监控时间内未完成, 设定重复次数。	1 到 15	用户
	重试结果	存储重复处理的次数。		
Ⓔ+8	到达 WDT 时间	READ 指令提示的读取操作的监控时间以秒为单位设定。 如果读取操作在设定时间内完成, 将按照Ⓔ+7 指定的次数传送重试。	1 到 32767 0 : 10 秒 固定的	用户
Ⓔ+9	接收数据长度	设定要读取的数据时钟的数量	1 到 480 *4	用户
Ⓔ+10	虚拟	未使用	—	—
Ⓔ+11	时钟设定标记 (错误时设定)	存储时钟数据是否接受状态 时钟数据禁止 : 0 时钟数据不允许 : 1	—	系统
Ⓔ+12	时钟数据 (错误时设定)	高 8 位 : 月 (1 到 12), 低 8 位 : 年 (0 到 99)	—	系统
Ⓔ+13		高 8 位 : 时 (0 到 23), 低 8 位 : 日 (1 到 31)		
Ⓔ+14		高 8 位 : 秒 (0 到 59), 低 8 位 : 分 (0 到 59)		
Ⓔ+15		高 8 位 : 00h, 低 8 位 : 星期 (0 到 6)		
Ⓔ+16	检测到错误的网络号	存储检测到错误的站点网络号 1 到 239 : 网络号	—	系统
Ⓔ+17	检测到错误的站点号	存储检测到错误的站点号 如果Ⓔ+1 结束状态是“通道使用中”, 则不存储数据。 1 到 64 : 站点号	—	系统

要点
(1) *3 : Jn 的设定值为 254 时指定。
(2) *4 : 如果 (⑨指定的软元件号) + (接收数据长度) 超过了⑨指定的软元件范围, 那么超出范围的数据点数将在下一个软元件开头处存储。

备注

- *1 : 关于故障代码请参照下列手册。
QnA/Q4AR MELSECNET/10 网络系统参考手册
- *2 : 关于网络模块的通道信息请参照第 8 章

关于 CC-Link, A(1S)J71QC24

CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

软元件	条目	数据设定	设定范围	设定者
⑨+0	执行类型	到达确认(位 0=1; 固定)	1	用户
⑨+1	结束状态	存储指令结束状态 0 : 没有错误 (正常结束) 不是 0 : 错误代码	—	系统
⑨+2	自站通道	自站使用的指定通道 1: 使用 CH1 2: 使用 CH2	1, 2	用户
⑨+3	虚拟	未使用	—	—
⑨+4	目标站网络号	固定为 0	0	用户
⑨+5	目标站号	固定为 0	0	用户
⑨+6	主/自站号	0: 指定主站 1 到 64: 指定自站	0 到 64	用户
⑨+7	传送重试次数	如果 READ 指令提示的读取操作在⑨+8 指定的监控时间内未完成, 设定重复次数。	0 到 15	用户
	重试结果	存储重复处理的次数。	—	系统
⑨+8	到达 WDT 时间	READ 指令提示的读取操作的监控时间以秒为单位设定。 如果读取操作在设定时间内完成, 将按照⑨+7 指定的次数传送重试。	1 到 32767 0 : 10 秒 固定的 (默认值)	用户
⑨+9	接收数据长度	设定要读取的数据时钟的数量	1 到 480 *	用户
⑨+10	虚拟	未使用	—	—
⑨+11	虚拟	未使用	—	—
⑨+12	虚拟	未使用	—	—
⑨+13				
⑨+14				
⑨+15				
⑨+16	检测到错误的网络号	固定为 0	—	系统
⑨+17	检测到错误的站点号	固定为 0	—	系统

要点
* : 如果 (⑨指定的软元件号) + (接收数据长度) 超过了⑨指定的软元件范围, 那么超出范围的数据点数将从下一个软元件开始存储。

[功能]

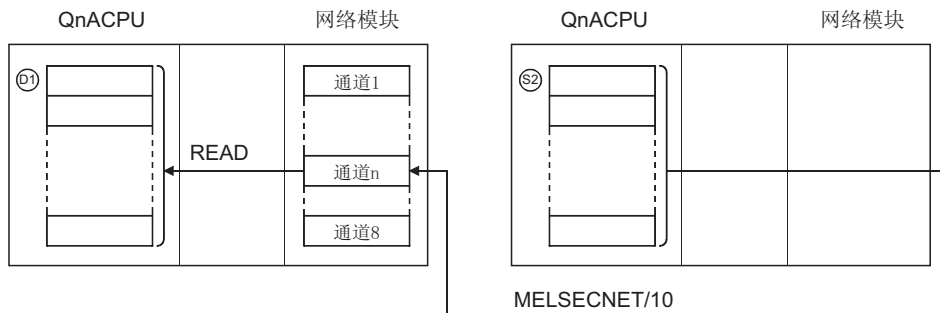
关于 MELSECNET/10, A(1S)J71QE71

- (1) 把存储在连接到 MELSECNET/10 网络的某站（由网络号和站号所指定）上的从⑤2指定的软元件开始的软元件中的数据，保存到本站上从⑤1指定的软元件开始的软元件中。

当目标站中软元件数据的读取操作已经完成时，被⑤2指定的结束软元件将会变为 ON。

[本站]

[目标站]



- (2) 除了连接在自站网络中的站点外，连接到 MELSECNET/10 网络并且站点号被指定的站点也可以执行软元件数据读取操作。

- (3) 数据链接指令不能在同一通道中对一个以上的地址执行。

如果具备了在两个或两个以上的地址同时执行指令的条件，那么自动收发关系（握手）将自动建立，因此以后将不能进行数据链接指令处理。

- (4) 使用通道的通讯指示标记，结束软元件⑤1和结束状态显示软元件 (⑤2+1) 可以检测 READ 指令的执行状态及其正常或错误结束。

(a) 通信指示标记

: READ 指令执行时为 ON，在 READ 操作结束时执行的 END 指令扫描处理时变为 OFF。

(b) 自站结束软元件

: 通过 READ 指令进行的读取操作结束，进行 END 指令扫描处理时变为 ON，并在进行下一个 END 指令处理时变为 OFF。

(c) 结束时的状态显示软元件

: 根据 READ 操作结束时的状态，变为 ON 或 OFF。

- 正常结束 : 保持状态不变
- 错误结束 : READ 指令操作结束时，执行 END 指令扫描处理时变为 ON，并在进行下一个 END 指令处理时变为 OFF。

关于 CC-Link, A(1S)J71QC24

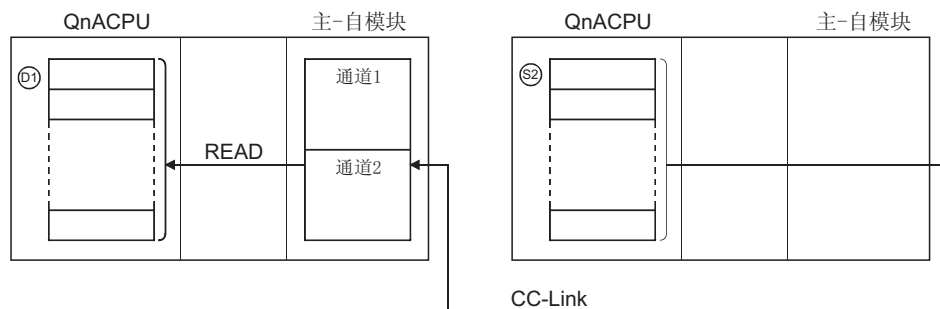
CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

- (1) 从与主/自站连接的由控制数据网络号指定站中被 $\textcircled{S2}$ 指定的字软元件开始存储软元件数据到的被本站 $\textcircled{S1}$ 指定的软元件开始的软元件中去。

当目标站中软元件数据的读取操作已经完成时, 被 $\textcircled{S2}$ 指定的结束软元件将会变为 ON。

[本站]

[目标站]



- (2) 软元件数据读操作可由相应的连接到 CC-Link 的站点进行。

- (3) 数据链接指令不能在同一通道的多个地址执行。

当在两个或两个以上地址同时执行的条件建立时, 将自动执行握手指令, 以使随后的数据链指令处理不能执行。

- (4) 对读指令的执行状态及正常结束还是错误结束可以由使用通道的通讯指示标记检测, 结束软元件 $\textcircled{S2}$ 及结束软元件($\textcircled{S2} + 1$)的状态显示。

(a) 通讯指示标志

: READ 指令执行过程中为 ON, 在读操作结束进行 END 指令扫描处理时为 OFF。

(b) 自站结束软元件

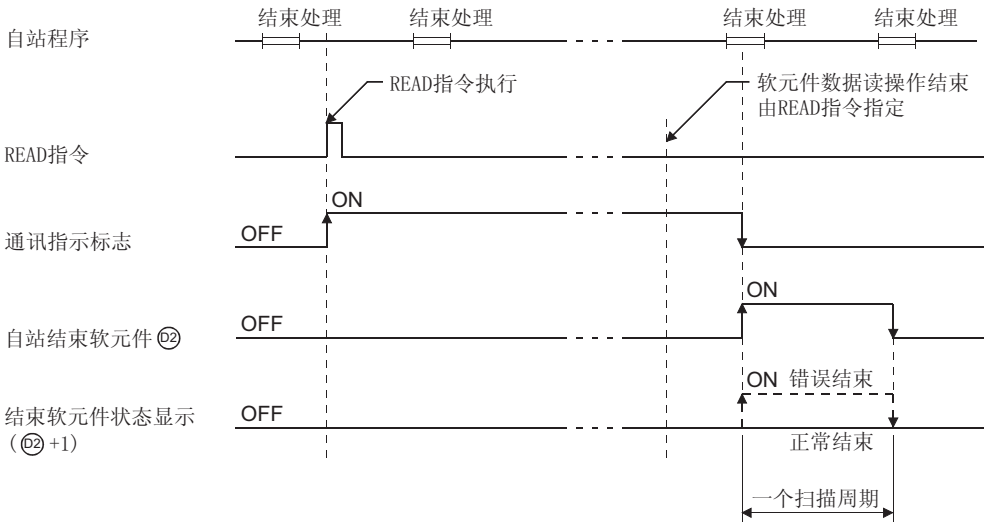
: 通过指令 READ 进行读取操作时, 进行 END 扫描处理变为 ON。进行下一个 END 指令处理时又变为 OFF。

(c) 结束时状态显示软元件

: 根据读指令结束时的状态打开或关闭。

- 正常结束 : 保持 OFF 状态不变
- 错误结束 : READ 指令结束, 进行 END 指令扫描处理时为 ON。进行下一个 END 指令处理时为 OFF。

[READ 指令执行期间自站运行]



[运行错误]

- (1) 在下列发生运行错误的情况中, 错误标志 (SM0) 变为, 并且错误代码存储在 SD0 中。
- 控制数据内容不在设定范围内 (错误代码: 4100)
 - Jn 指定的网络号没有连接到这个站点 (错误代码: 4102)
 - Un 指定的 I/O 号模块不是网络模块 (错误代码: 2111)

备注

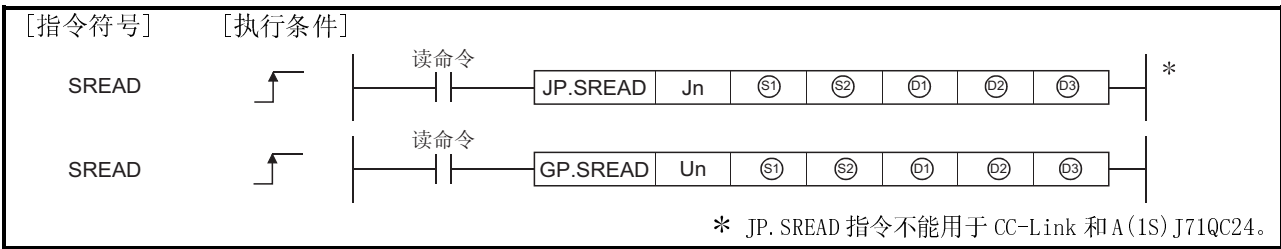
1)* : 通道使用的通讯指示标记说明如下:
对 CC-Link 和 A(1S)J71QC24 最多两个通道有效

通道号	1	2	3	4	5	6	7	8
通讯指示标记号	SB30	SB32	SB34	SB36	SB38	SB3A	SB3C	SB3E

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×*4	×*4	×*4	○	○

8.2.2 从其它站点读取字软元件数据 (SREAD)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[]\C[]		特殊功能模块 U[]\G[]	变址寄存器 Zn	常数	其它
	位	字		位	字				
(S1)	—	○				—			
(S2)	—	○				—			
(D1)	—	○				—			
(D2)	○	○				—			
(D3)	○	○				—			



[设定数据]

用于 MELSECNET/10, A(1S)J71QE71

设定数据	含义	数据类型
Jn	自站网络号 *1	BIN 16 位
Un	网络模块自站起始 I/O 号 *2	
Ⓔ1	存储控制数据的自站的起始软元件	软元件名
Ⓔ2	存储读取数据的站点的起始软元件	
Ⓔ1	存储读取数据的自站的起始软元件	
Ⓔ2	指令结束后将要变为 ON 一个扫描周期的自站软元件 *3	位
Ⓔ3	指令结束后将要变为 ON 一个扫描周期的目标站软元件 *3	

备注

- *1 : 自站网络号按如下指定
 - 1 到 239 : 网络号
 - 254 : 由设定为其它站可以访问的有效模块来指定的网络
- *2 : 自站网络模块的起始 I/O 号的指定范围为 0 到 FEH。
- *3 : 本地软元件和每个程序的文件寄存器不能用作数据设定软元件 (详细内容参考 QnACPU 编程手册 (特殊功能模块)。)
- *4 : QCPU 可以使用的 SREAD 指令的详细内容, 请参考以下手册。
 - Q 相应的 MELSECNET/H 网络系统参考手册 (PLC 到 PLC 网络)。
 - Q 相应的以太网接口模块用户手册 (应用篇)。

要点

- (1) 只有目标站点是 QCPU/QnACPU 时才可以执行 SREAD 指令。
(ACPU 连接到 MELSECNET/10 时, 不能执行 SREAD 指令。)
只可以把 QCPU/QnACPU 站点号设定为目标站点号。
- (2) 用 SREAD 指令从其它站点的 CPU 模块软元件中读取数据时, 在可应用于自站 CPU 模块的范围内指定软元件。
(在其它站点 CPU 模块读取的起始软元件号Ⓔ2) + (读取点数-1) ≤ (自站 CPU 模块的最后软元件号*)
*: 自站 CPU 模块的最后软元件号的名与Ⓔ2相同。

关于 CC-Link, A(1S)J71QC24

CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

设定数据	含义	数据类型
Un	主/本模块自站的起始 I/O 号 *1	BIN 16 位
Ⓔ①	存储控制数据的自站的起始软元件	软元件名
Ⓔ②	存储读取数据的站的起始软元件	
Ⓔ①	存储读取数据的自站的起始软元件	
Ⓔ②	指令结束后将要变为 ON 一个扫描周期的自站软元件 *3	
Ⓔ③	指令结束后将要变为 ON 一个扫描周期的目标站软元件 *2	位

备注

*1 自站的主/自站模块起始 I/O 号的指定范围为从 0 到 FEh。

*2 每个程序的本地软元件和文件寄存器不能用作设定数据的软元件。
(详细资料见 QnACPU 编程手册(特殊功能模块))

[控制数据]

用于 MELSECNET/10, A(1S)J71QE71

软元件	条目	设定数据	设定范围	设定者
Ⓔ①+0	执行类型	到达确定 (位 0=1; 固定)	0001h 0081h	用户
	出错完成类型	错误处理完成后设定时钟数据设定状态 不设定时钟数据 : 将位 7 (b7) 设定成 0 设定时钟数据 : 将位 7 (b7) 设定成 1 (Ⓔ①+11 在前面添加)		
Ⓔ①+1	完成状态	指令完成被存储后的状态 0 : 没有错误 (正常完成) 不是 0 : 错误代码 *1	—	系统
Ⓔ①+2	自站通道	指定自站将使用的通道 *2	1 到 8	用户
Ⓔ①+3	虚拟	未使用	0	—
Ⓔ①+4	目标站点网络号	设定指定读取软元件数据站点的网络号	1 到 239 254 *3	用户
Ⓔ①+5	目标站点号	设定指定站点的站号	1 到 64	用户
Ⓔ①+6	虚拟	未使用	—	—
Ⓔ①+7	传送重试次数	在Ⓔ①+8 指定的 WDT 时间范围内通过 SREAD 指令的读写操作未完成, 设定传送重试次数	1 到 15	用户
	重试结果	重复次数被存储。		
Ⓔ①+8	到达 WDT 时间	通过 SREAD 指令设定读操作的 WDT 时间的设定以秒为单位。 如果读操作在设定时间内未完成, 将按照Ⓔ①+7 指定的次数重试发送。	1 到 32767 0 : 固定 10 秒	用户
Ⓔ①+9	接收数据长度	设定要读取的数据块的数量	1 到 480 *4	用户
Ⓔ①+10	虚拟	未使用	—	—
Ⓔ①+11	时钟设定标记 (只在出错时设定)	存储时钟数据允许/禁止状态 时钟数据禁止 : 0 时钟数据允许 : 1	—	系统
Ⓔ①+12	时钟数据 (只在出错时设定)	高 8 位: 月份 (1 到 12), 低 8 位: 年份 (0 到 99)	—	系统
Ⓔ①+13		高 8 位: 时 (0 到 23), 低 8 位: 日 (1 到 31)		
Ⓔ①+14		高 8 位: 秒 (0 到 59), 低 8 位: 分 (1 到 59)		
Ⓔ①+15		高 8 位: 00h, 低 8 位: 星期 (0 到 6)		
Ⓔ①+16	检测到错误的网络号	低 8 位: 星期 (0 到 6) 1 到 239 : 网络号	—	系统
Ⓔ①+17	检测到错误的网络号	存储发现错误的站点号 但当Ⓔ①+1 完成状态是“通道使用中”时不存储数据。 1 到 64 : 站点号	—	系统

要点
(1) *3 : 当值 254 已被 Jn 设定时指定。 (2) *4 : 如果 (⑥指定的软元件号) + (接收数据长度) 超过了⑥指定的软元件号, 超过指定范围的数据点数将从下一个软元件开始被存储。

备注

- 1) *1 : 错误代码参考下列手册。
 QnA/Q4AR MELSECNET/10 网络系统参考手册
- 2) *2 : 网络模块通道资料见第 8 章。

关于 CC-Link, A(1S)J71QC24

CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

软元件	类型	设定数据	设定范围	设定者
⑤+0	执行类型	到达确定 (位 0=1; 固定)	1	用户
⑤+1	完成状态	指令完成被存储后的状态 0 : 没有错误 (正常完成) 不是 0 : 错误代码	—	系统
⑤+2	自站通道	指定自站将使用的通道 1: 使用 CH1 侧 2: 使用 CH2 侧	1, 2	用户
⑤+3	虚拟	未使用	—	—
⑤+4	目标站点网络号	固定为 0	0	用户
⑤+5	目标站点号	固定为 0	0	用户
⑤+6	主/本机站号	0: 指定主站 1 到 64: 指定本机站	0 到 64	用户
⑤+7	传送重试次数	在⑤+8 指定的 WDT 时间范围内通过 SREAD 指令的读写操作未完成, 设定传送重试次数	0 到 15	用户
	重试结果	重复次数被存储。	—	系统
⑤+8	到达 WDT 时间	通过 SREAD 指令设定的读写操作的 WDT 时间的设定以秒为单位。 如果读写操作在设定时间内未完成, 将按照⑤+7 指定的次数重试发送。	1 到 32767 0 : 固定 10 秒 (默认)	用户
⑤+9	接收数据长度	设定要读取的数据块的数量	1 到 480 *	用户
⑤+10	虚拟	未使用	—	—
⑤+11	虚拟	未使用	—	—
⑤+12	虚拟	未使用	—	—
⑤+13				
⑤+14				
⑤+15				
⑤+16	检测到错误的网络号	固定为 0	—	系统
⑤+17	检测到错误的站点号	固定为 0	—	系统

要点
* : 如果 (⑥指定的软元件号) + (接收数据长度) 超过了⑥指定的软元件号, 那么超出范围的数据点数将从下一个软元件开始存储。

[功能]

[用于 MELSECNET/10, A(1S)J71QE71]

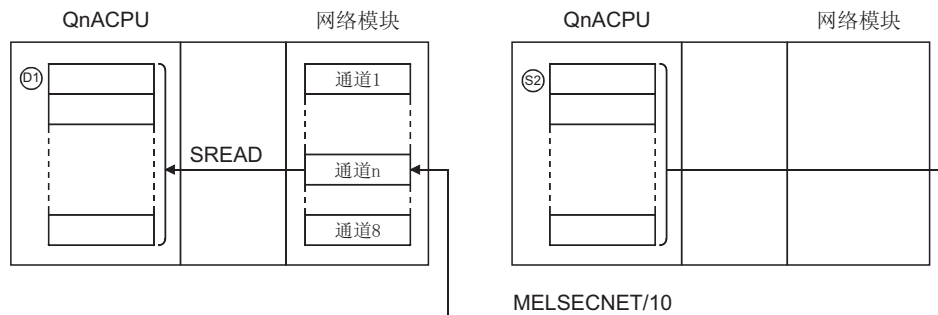
- (1) 把存储在连接到 MELSECNET/10 网络的某站（由网络号和站号所指定）上的从②指定的软元件开始的软元件中的数据，保存到自站上从③指定的软元件开始的软元件中。

当目标站中软元件数据的读取操作已经完成时，被④指定的结束软元件将会变为 ON。

同样，当被②指定的软元件数据传送已经完成时，被③指定的软元件在目标站中将会变为 ON。

[本站]

[目标站]



- (2) 除了连接到自站网络中的站之外，软元件数据的写入操作能够在连接到指定网络号的 MELSECNET/10 网络中的站上执行。

- (3) 在同一个通道中，数据链接指令不能在多个位置上同时执行。

当在两个或更多位置同时执行的条件已经建立时，收发关系（握手）将被自动执行，所以后续处理中将不能执行数据链接指令。

- (4) SREAD 指令的执行状态及其正常或错误结束可以通过使用通道的通讯指示标记，结束软元件（②和③）和结束软元件的状态显示（④+1）来检测。

(a) 通讯指示标志

：在 SREAD 指令执行时变为 ON，在读取操作完成时的扫描周期的 END 指令处理为 OFF 时变为 OFF。

(b) 自站结束软元件

：通过 SREAD 指令执行的读取操作结束时的扫描周期中被 END 处理变为 ON，被下一个 END 处理时变为 OFF。

(c) 结束时的状态显示软元件

：按照 SREAD 指令结束时的状态变为 ON 或 OFF。

- 正常结束 ：保持 OFF 状态不改变
- 错误结束 ：当 SREAD 指令已经结束时，在扫描周期的 END 处理时变为 ON，在下一个 END 处理时变为 OFF。

(d) 目标站的结束软元件

：当由 SREAD 指令指定的软元件的数据传送已经结束时，在扫描周期的 END 处理时变为 ON，在下一个 END 处理时变为 OFF。

[关于 CC-Link, A(1S)J71QC24]

CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

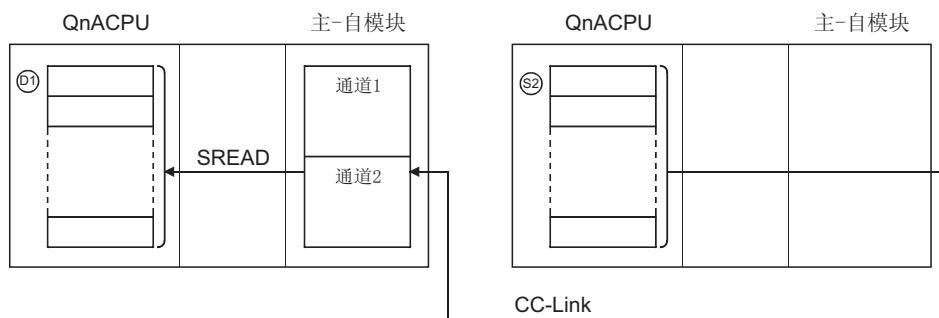
- (1) 从与主/自站连接的由控制数据网络号指定站中被⑤②指定的字软元件开始存储软元件数据到的被本站⑤③指定的软元件开始的软元件中去。

当目标站的软元件数据读取操作已经结束, 被⑤②指定的结束软元件变为 ON。

同时, 当被⑤②指定的软元件数据传送已经结束, 在目标站被⑤③指定的软元件变为 ON。

[本站]

[目标站]



- (2) 软元件数据读取操作能够在用 CC-Link 连接的站中进行。

- (3) 在同一个通道中, 数据链接指令不能在多于一个位置上执行。

当在两个或更多位置同时执行的条件已经建立时, 建立收发关系(握手)将被自动执行, 所以后续处理中将不可能有数据链接指令。

- (4) SREAD 指令的执行状态和是否为正常结束还是异常结束能够通过使用通道的通讯指示标志*, 结束软元件(⑤②和⑤③)和结束软元件的状态显示(⑤③+1)来检查。

- (a) 通讯指示标志

: 在 SREAD 指令执行时变为 ON, 在读取操作完成后的扫描周期的 END 指令处理为 OFF 时变为 OFF。

- (b) 自站结束软元件

: 通过 SREAD 指令执行的读取操作结束的扫描周期中被 END 处理变为 ON, 被下一个 END 处理变为 OFF

- (c) 结束时的状态显示软元件

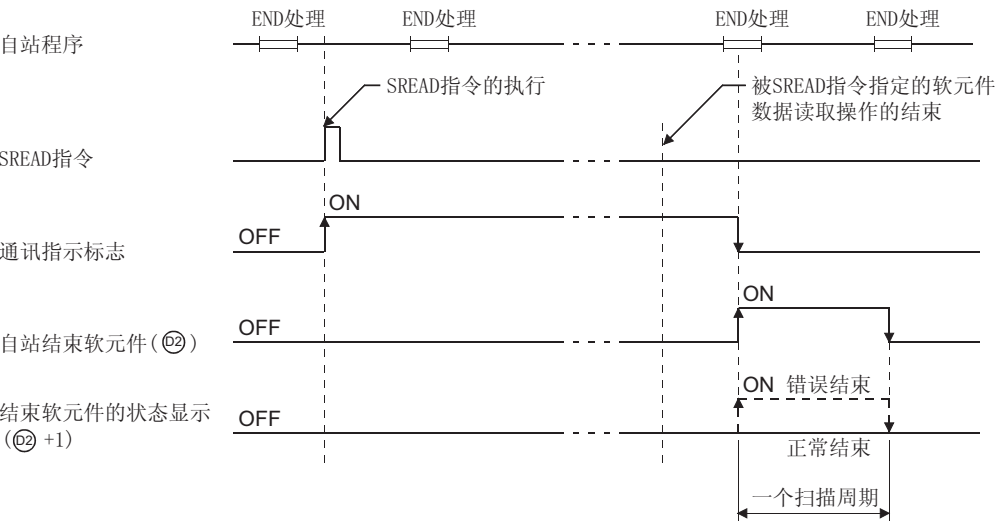
: 按照 SREAD 指令结束时的状态变为 ON 或变为 OFF。

- 正常结束 : 保持 OFF 状态, 不改变
- 出错结束 : 当 SREAD 指令已经结束时, 在扫描周期的 END 处理时变为 ON, 在下一个 END 处理时变为 OFF。

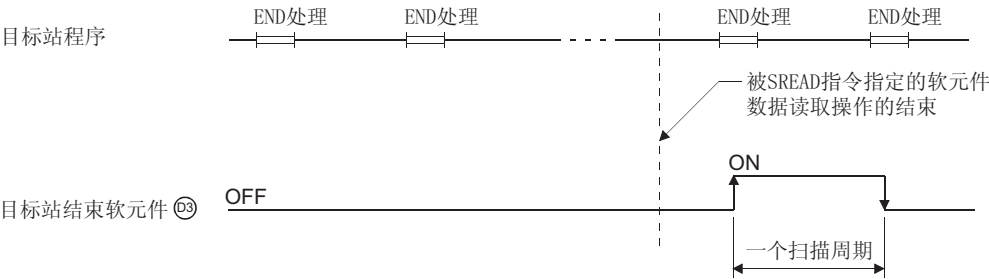
- (d) 目标站的结束软元件

: 当被 SREAD 指令指定的软元件数据传送已经结束, 该软元件在扫描周期的 END 处理时被变为 ON, 并且在下一个 END 处理中被变为 OFF。

[当执行 SREAD 指令时的自站操作]



[当执行 SREAD 指令时的目标站的操作]



备注

- 1) *:使用通道的通讯指示标志列示如下：用于 CC-Link 和 A(1S)J71QC24，最多可以使用两个通道。

通道号	1	2	3	4	5	6	7	8
通讯指示标志号	SB30	SB32	SB34	SB36	SB38	SB3A	SB3C	SB3E

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×*4	×*4	×*4	○	○

8.2.3 数据写入到 MELSECNET/10 网络中的站(WRITE)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[]\[]		特殊功能模块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
①	—	○				—	
②	—	○				—	
③	—	○				—	
④	○	○				—	

[指令符号]

WRITE

↑

写入指令

JP.WRITE

Jn

①

②

③

④

*

WRITE

↑

写入指令

GP.WRITE

Un

①

②

③

④

*

* JP.WRITE 指令不能用于 CC-Link 和 A(1S) J71QC24。

[设定数据]

[用于 MELSECNET/10, A(1S) J71QE71]

设定数据	含义	数据类型
Jn	本站网络号 *1	BIN 16 位
Un	网络模块本站起始 I/O 号 *2	
①	存储控制数据的本站起始软元件	软元件名称
②	写入数据正在被存储的本站起始软元件	
③	目标站的起始软元件, 该目标站中被写入数据将要被存储	
④	指令完成后在一个扫描周期内变为 ON 的软元件 *3	位

备注

- *1: 本站网络号按如下指定:
 - 1 到 239 : 网络号
 - 254 : 由设定为其它站可以访问的有效模块来指定的网络
- *2: 本站网络模块的起始 I/O 号的指定可以为 0 到 FEH。
- *3: 每一个程序的局部软元件和文件寄存器不能当作设定数据使用的软元件来使用。(详细资料请参见 QnACPU 编程手册(特殊功能模块))
- *4: 能够在 QCPU 中使用的 WRITE 指令的详细资料, 参见以下手册:
 - Q 系列 MELSECNET/H 网络系统参考手册(PLC 到 PLC 网络)
 - Q 系列以太网接口模块用户手册(应用篇)

要点	
(1)	只有当目标站为 QCPU/QnACPU 站时，WRITE 指令才能被执行。 (WRITE 指令不能在连接到 MELSECNET/10 网络的 ACPU 中执行。) 只有 QCPU/QnACPU 站号才应被设定为目标站号。
(2)	“FFH(在目标网络中的所有站)”能够通过 WRITE 指令指定到只由 QnACPU 站构成的网络的目标站的站号中去。 由 QnACPU 和 ACPU 站混合构成的网络中不能有“FFH”指定。
(3)	当用 WRITE 指令从其它站 CPU 模块软元件中写入数据时，在适用范围内指定软元件到自站 CPU 模块。 (在其它站 CPU 模块②中待写入读取的起始软元件号) + (写入数号 - 1) ≤ (自站 CPU 模块的最后一个软元件号*) *: 自站 CPU 模块的最后一个软元件号和②有相同的软元件名称。

[关于 CC-Link, A(1S)J71QC24]

CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

设定数据	含义	数据类型
Un	主/自站的自站起始 I/O 号 *1	BIN 16 位
①	存储控制数据的自站起始软元件	软元件名称
②	写入数据存储自站的起始软元件	
③	目标站的起始软元件，该目标站中的数据在写入时将被存储	
④	指令结束时，在一个扫描周期内变为 ON 的软元件 *2	位

REMARKS

- *1: 自站主/自模块的起始 I/O 号的指定可以为 0 到 FEH。
 *2: 每一个程序的局部软元件和文件寄存器均不能作为设定数据使用的软元件来使用。
 (详情请参考 QnACPU 编程手册(特殊功能模块))。

[控制数据]

[用于 MELSECNET/10, A(1S)J71QE71]

软元件	项目	设定数据	设定范围	设定者
Ⓔ + 0	执行类型	传送结束确认 : 设定位 0 为 0 无传送结束确认 : 设定位 0 为 1	0000 _h	用户
	错误结束类型	当出错处理结束时设定时钟数据设定状态 不设定时钟数据 : 设定位 7 (b7) 为 0 设定时钟数据 : 设定位 7 (b7) 为 1 (Ⓔ+10 向前附加)	0001 _h 0080 _h 0081 _h	
Ⓔ + 1	结束状态	在指令结束时的状态被存储 0 : 无出错 (正常结束) 除 0 之外 : 错误代码 *1	—	系统
Ⓔ + 2	自站使用的通道	指定自站将要使用的通道 *2	1 到 8	用户
Ⓔ + 3	虚拟	不使用	0	—
Ⓔ + 4	目标站的网络号	为目标站指定网络号	1 到 239 254 *3	用户
Ⓔ + 5	目标站号	为指定的站设定站号	1 到 64: 站号 81 _h 到 89 _h : 组指定 FF _h : 所有在目标网络 中的站	用户
Ⓔ + 6	虚拟	不使用	—	—
Ⓔ + 7	传送重试次数	当被Ⓔ指定的执行类型为 1 时有效, 当 WDT 被Ⓔ+8 指定的时间内没有完成结束时, 设定将要进行的传送重试次数。	0 到 15	用户
	重试结果	执行重复次数被存储		系统
Ⓔ + 8	到达 WDT 时间	指令结束的 WDT 时间是以秒为单位设定的。如果在 WDT 设定的时间内操作还没有结束时, 按照被Ⓔ+7 指定的次数传送被重试。	1 到 32767 0 : 10 秒固定 只当执行类型在 1 处被设定时有效。	用户
Ⓔ + 9	传送数据长度	设定将被写入的数据块数目	1 到 480	用户
Ⓔ + 10	虚拟	不使用	—	—
Ⓔ + 11	时钟设定标志 (只在出错时设定)	S 存储时钟数据的允许禁止状态 当时钟数据禁止 : 0 当时钟数据允许 : 1	—	系统
Ⓔ + 12	时钟数据 (只在出错时设定)	高 8 位 : 月 (1 到 12), 低 8 位 : 年 (0 到 99)	—	系统
Ⓔ + 13		高 8 位 : 时 (0 到 23), 低 8 位 : 日 (1 到 31)		
Ⓔ + 14		高 8 位 : 秒 (0 到 59), 低 8 位 : 分 (0 到 59)		
Ⓔ + 15		高 8 位 : 00 _h , 低 8 位 : 星期 (0 到 6)		
Ⓔ + 16	检测到错误的网络号	存储检测到错误的站的网络号。 1 到 239 : 网络号	—	系统
Ⓔ + 17	检测到错误的站号	存储检测到错误站的站号。 如果Ⓔ+1 结束状态为“通道使用中”时, 不存储数据。 1 到 64 : 站号	—	系统

备注

*1: 关于错误代码, 请参考下列手册:

QnA/Q4AR 用 MELSECNET/10 网络系统参考手册

*2: 网络模块通道的信息, 请参见第 8 章。

*3: 当 Jn 被设定为 254 时指定。

[关于 CC-Link, A(1S)J71QC24]

CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

软元件	项目	设定数据	设定范围	设定者
⑨1 + 0	执行类型	0: 无到达确认 1 (执行到达确认)	0, 1	用户
⑨1 + 1	结束状态	指令结束时状态被存储 0 : 无出错 (正常结束) 除 0 之外 : 错误代码	—	系统
⑨1 + 2	自站使用的通道	指定自站将要使用的通道 1: 使用 CH1 通道 2: 使用 CH2 通道	1, 2	用户
⑨1 + 3	虚拟	不使用	—	—
⑨1 + 4	目标站的网络号	固定为 0	0	用户
⑨1 + 5	目标站号	固定为 0	0	用户
⑨1 + 6	主站/自站号	0: 指定的主站 1 到 64: 指定的自站	0 到 64	用户
⑨1 + 7	传送重试次数	当被⑨1指定的执行类型为 1 时有效, 并当 WDT 中在被⑨1+8 指定的时间内没有完成结束, 设定将要进行的传送重试次数	0 到 15	用户
	重试结果	存储重复处理的次数。	—	系统
⑨1 + 8	到达 WDT 时间	指令结束的 WDT 时间是以秒为单位设定的。如果操作在为 WDT 设定的时间内还没有结束, 那么传送按照被⑨1+7 指定的次数被重复。	1 到 32767 0: 10 秒固定 (缺省) 仅当执行类型设定为 1 时有效。	用户
⑨1 + 9	传送数据长度	设定将要被写入的数据块的数目。	1 到 480	用户
⑨1 + 10	虚拟	不使用	—	—
⑨1 + 11	虚拟	不使用	—	—
⑨1 + 12	虚拟	不使用	—	—
⑨1 + 13				
⑨1 + 14				
⑨1 + 15				
⑨1 + 16	检测到错误的网络号	固定为 0	—	系统
⑨1 + 17	检测到错误的站号	固定为 0	—	系统

[功能]

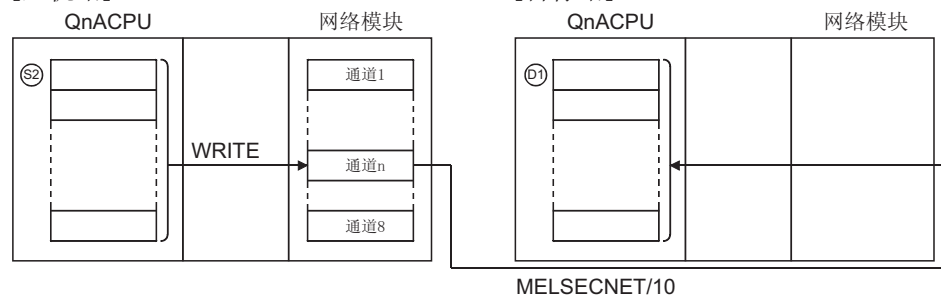
[用于 MELSECNET/10, A(1S)J71QE71]

- (1) 把从自站⑤指定的软元件开始存储的软元件数据存储在到与 MELSECNET/10 网络连接的某站（由网络号和站号所指定）中从⑥指定的字软元件开始的字软元件中去。

当目标站中的写入操作已经结束后，被⑥指定的结束软元件将会变为 ON

[主机站]

[目标站]



- (2) 除了连接到自站网络的站之外，软元件数据的写入操作也能够连接到 MELSECNET/10 指定的网络号中的站上执行。

- (3) 在同一个通道中，数据链接指令不能在多个位置上执行。

当在两个或更多位置同时执行的条件已经建立时，建立收发关系（握手）将被自动执行，所以后续处理中将不能执行数据链接指令。

- (4) WRITE 指令是在后台执行的，该指令的正常和错误结束可以通过正在使用的通道的通讯指示标志(结束软元件⑥和自站的结束软元件(⑥+1)状态显示)来确认。

(a) 通讯指示标志

：当 WRITE 指令执行时为 ON，当写入操作已经结束时，在扫描周期的 END 处理时变为 OFF。

(b) 自站结束软元件

：当 WRITE 指令已经结束时，在扫描周期的 END 处理时变为 ON，在下一个 END 处理时变为 OFF。

(c) 结束时的状态显示软元件

：变为 ON 或 OFF 取决于 WRITE 指令结束时的状态。

- 正常结束 ：保持 OFF 状态不变。
- 出错结束 ：当 WRITE 指令已经结束时在扫描周期的 END 处理时为 ON，在下一个 END 处理时变为 OFF。

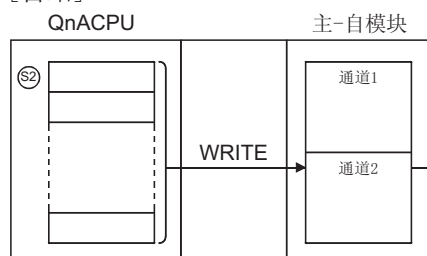
[关于 CC-Link, A(1S)J71QC24]

CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

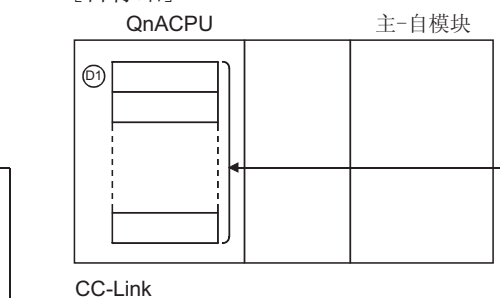
- (1) 把从自站②指定的软元件开始存储的软元件数据存储在到与 MELSECNET/10 网络连接的某站（由网络号和站号所指定）中从②指定的字软元件开始的字软元件中去。

当目标站中写入操作已经完成时，被②指定的结束软元件将会变为 ON。

[自站]



[目标站]



- (2) 当软元件数据的写入操作能够在与 CC-Link 连接的站中执行。

- (3) 在同一个通道中，数据链接指令不能在多个位置上执行。

当在两个或更多位置同时执行的条件已经建立时，建立收发关系（握手）将被自动执行，所以后续操作中不能有数据链接指令。

- (4) WRITE 指令是在后台执行的，该指令是正常或错误结束能够通过正在使用的通道的通讯指示标志(结束软元件②和自站中结束软元件(②+1)的显示状态)来确认。

(a) 通讯指示标志

: 在 WRITE 指令执行时为 ON，当写入操作已经结束时，在扫描周期的 END 处理时变为 OFF。

(b) 自站结束软元件

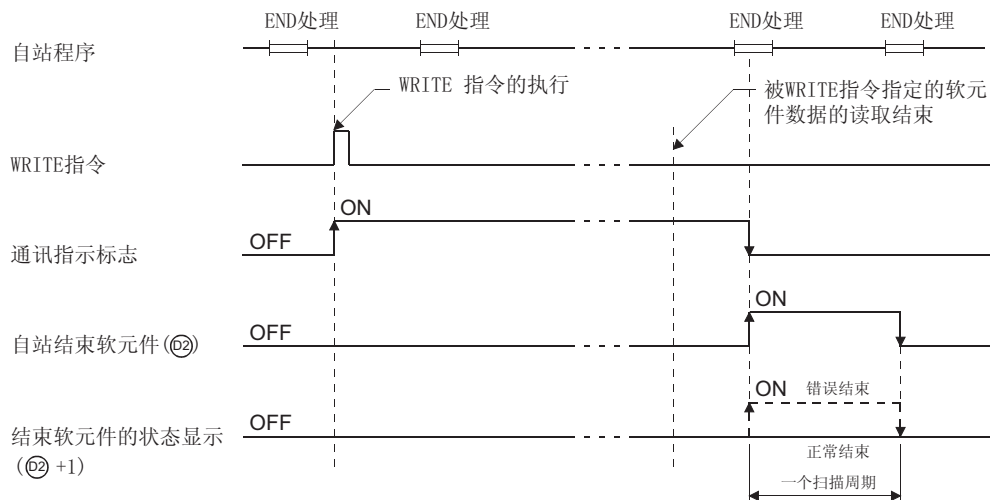
: 当 WRITE 指令已经结束时，在扫描周期的 END 处理时变为 ON，在下一个 END 处理时变为 OFF。

(c) 结束时的状态显示软元件

: 变为 ON 或 OFF 取决于 WRITE 指令结束时的状态

- 正常结束 : 保持 OFF，不改变
- 出错结束 : 当 WRITE 指令已经结束时在扫描周期的 END 处理时变为 ON，在下一个 END 处理时变为 OFF。

[在 WRITE 指令执行过程中的自站操作]



[运行错误]

(1) 在以下发生运行错误情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0。

- 控制数据内容不在设定范围中(错误代码: 4100)。
- 被 Jn 指定的网络号没有连接到该站(错误代码: 4102)。
- 被 Un 指定的 I/O 号模块不是网络模块(错误代码: 2111)。

备注

1) *:被使用的通道的通讯指示标志列示如下:

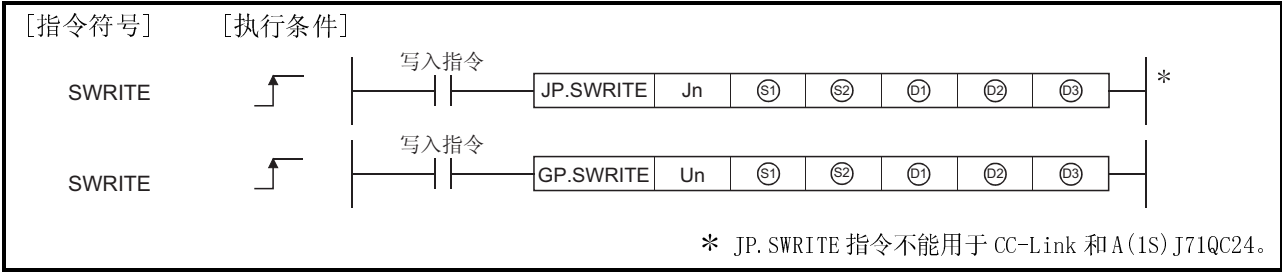
用于 CC-Link 和 A(1S)J71QC24, 最多有两个通道可以使用。

通道号	1	2	3	4	5	6	7	8
通讯指示标志号	SB30	SB32	SB34	SB36	SB38	SB3A	SB3C	SB3E

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×*4	×*4	×*4	○	○

8.2.4 写入软元件数据到其它站 (SWRITE)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[][]		特殊功能模块 U[][G][]	变址寄存器 Zn	常数	其它
	位	字		位	字				
(S1)	—	○				—			
(S2)	—	○				—			
(D1)	—	○				—			
(D2)	○	○				—			
(D3)	○	○				—			



[设定数据]

[用于 MELSECNET/10, A(1S)J71QE71]

设定数据	含义	数据类型
Jn	自站网络号 *1	BIN 16 位
Un	网络模块的自站起始 I/O 号*2	
S1	存储控制数据的自站起始软元件	软元件名称
S2	写入数据存储的自站起始软元件	
D1	写入时存储数据的目标站起始软元件	
D2	当指令已经结束时在一个扫描周期内变为 ON 的自站软元件 *3	位
D3	当指令已经结束时在一个扫描周期内变为 ON 的目标站软元件 *3	

备注

- *1: 自站的网络号将被指定如下:
- 1 到 239 : 网络号

● 254 : 由设定为其它站可以访问的有效模块来指定的网络。
- *2: 自站网络模块的起始 I/O 号的指定可以为 0 到 FEH。
- *3: 每个程序的局部软元件和文件寄存器均不能当作为设定数据使用的软元件来使用 (详细情况请参见 QnACPU 编程手册 (特殊功能模块))
- *4: 可以在 QCPU 中使用的 WRITE 指令的详细情况, 请参见下列手册:
- Q 系列 MELSECNET/H 网络系统参考手册 (PLC 到 PLC 网络)

● Q 系列以太网接口模块用户手册 (应用篇)

要点	
	(1) 仅当目标站为一个 QCPU/QnACPU 站时，SWRITE 指令才能被执行。 (在连接到一个 MELSECNET/10 网络的 ACPU 站中，SWRITE 指令不能被执行) 只有 QCPU/QnACPU 的站号才能被设定为目标站号。
	(2) 指定“FFH (在对象网络中的所有站)”能够通过到网络中的一个目标站号的 SWRITE 指令来完成，该网络仅由 QnACPU 站构成。 在由 QnACPU 和 ACPU 站混合构成的网络中，“FFH”指定不能进行。
	(3) 当用 SWRITE 指令从其它站 CPU 模块软元件写入数据时，在自站 CPU 模块中适用的范围内指定软元件。 (在其它站 CPU 模块②中将要被写入的起始软元件号) + (写入的点数 - 1) ≤ (自站 CPU 模块的最后一个软元件号*) *: 自站 CPU 模块的最后一个软元件号和②具有相同的软元件名称。

[关于 CC-Link, A(1S)J71QC24]

CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

设定数据	含义	数据类型
Un	主/本模块自站的起始 I/O 号 *1	BIN 16 位
①	本站控制数据的起始软元件	软元件名称
②	站中存储了写入数据的起始软元件	
③	目标站的起始软元件，该目标站中的数据被写入时将被存储	
④	当指令已经结束时在一个扫描周期内变为 ON 的自站软元件	位
⑤	当指令已经结束时在一个扫描周期内变为 ON 的目标站软元件 *2	

备注

- *1: 自站主/自模块的起始 I/O 号的指定可以为 0 到 FFH。
*2: 每一个程序的局部软元件和文件寄存器均不能当作设定数据使用的软元件来使用。
(详细情况请参考 QnACPU 编程手册 (特殊功能模块)。)

[控制数据]

[用于 MELSECNET/10, A(1S)J71QE71]

软元件	项目	设定数据	设定数据	设定者
Ⓔ + 0	执行类型	无传送结束确认 : 设定位 0 为 0 传送结束确认 : 设定位 0 为 1	0000 _h	用户
	出错结束类型	当出错处理结束时设定时钟数据设定状态。 不设定时钟数据 : 设定位 7(b7) 为 0 设定时钟数据 : 设定位 7(b7) 为 1 (Ⓔ+10 向前附加)	0001 _h 0080 _h 0081 _h	
Ⓔ + 1	结束状态	指令结束时的状态被存储。 0 : 无错误(正常结束)。 除 0 之外 : 错误代码 *1	—	系统
Ⓔ + 2	自站使用的通道	指定自站将使用的通道 *2	1 到 8	用户
Ⓔ + 3	虚拟	不使用	0	—
Ⓔ + 4	目标站的网络号	为目标站指定网络号	1 到 239 254 *3	用户
Ⓔ + 5	目标站号	为目标站设定站号	1 到 64: 站号 81 _h 到 89 _h : 组指定 FF _h : 在目标网络中的 所有站	用户
Ⓔ + 6	虚拟	不使用	—	—
Ⓔ + 7	传送重试次数	当被Ⓔ指定的执行类型为 1 时有效, 并且当在 WDT 中被Ⓔ+8 指定的时间内没有结束时, 设定将要进行的传送重试次数。	0 到 15	用户
	重试结果	执行的重复次数被存储		系统
Ⓔ + 8	到达 WDT 时间	指令结束指令结束的 WDT 时间是以秒为单位设定的。如果在 WDT 设定的时间内操作还没有结束时, 传送按照被Ⓔ+7 指定的次数被重试	1 到 32767 0 : 10 秒固定 只当执行类型在 1 处被设定时有效。	用户
Ⓔ + 9	传送数据长度	设定将要写入的数据数目	1 到 480	用户
Ⓔ + 10	虚拟	不使用	—	—
Ⓔ + 11	时钟设定标志 (只在出错时设定)	存储时钟数据的允许/禁止状态 当时钟数据禁止 : 0 当时钟数据允许 : 1	—	系统
Ⓔ + 12	时钟数据 (只在出错时设定)	高 8 位 : 月 (1 到 12), 低 8 位 : 年 (0 到 99)	—	系统
Ⓔ + 13		高 8 位 : 时 (0 到 23), 低 8 位 : 日 (1 到 31)		
Ⓔ + 14		高 8 位 : 秒 (0 到 59), 低 8 位 : 分 (0 到 59)		
Ⓔ + 15		高 8 位 : 00 _h , 低 8 位 : 星期 (0 到 6)		
Ⓔ + 16	检测到错误的网络号	存储检测到错误的站的网络号。 1 到 239 : 网络号	—	系统
Ⓔ + 17	检测到错误的站号	存储检测到错误的站的站号。 如果Ⓔ+1 结束状态为“通道使用中”时, 不存储数据。 1 到 64 : 站号	—	系统

备注

*1: 关于错误代码, 请参考下列手册:

QnA/Q4AR 用 MELSECNET/10 网络系统参考手册

*2: 网络模块通道的信息, 请参见第 8 章。

*3: 当 Jn 被设定为 254 时指定。

[关于 CC-Link, A(1S)J71QC24]

CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

软元件	项目	设定数据	设定范围	设定者
⑨1 + 0	执行类型	0: 无到达确认 1 (执行到达确认)	0, 1	用户
⑨1 + 1	结束状态	指令结束时的状态被存储 0 : 无错误 (正常结束) 除 0 之外 : 错误代码	—	系统
⑨1 + 2	自站使用的通道	指定自站将使用的通道 1: 使用 CH1 通道 2: 使用 CH2 通道	1, 2	用户
⑨1 + 3	虚拟	不使用	—	—
⑨1 + 4	目标站的网络号	固定 0	0	用户
⑨1 + 5	目标站号	固定 0	0	用户
⑨1 + 6	主/自站号	0: 主站被指定 1 到 64: 自站被指定	0 到 64	用户
⑨1 + 7	传送重试次数	当被⑨指定的执行类型为 1 时有效, 当 WDT 被⑨+8 指定的时间内没有完成结束时, 设定将要进行的传送重试次数。	0 到 15	用户
	重试结果	执行的重复次数被存储	—	系统
⑨1 + 8	到达 WDT 时间	指令结束的 WDT 时间是以秒为单位设定的。如果在 WDT 设定的时间内操作还没有结束时, 按照被⑨+7 指定的次数传送被重试。.	1 到 32767 0: 10 秒固定 (缺省) 只当执行类型在 1 处被设定时有效。	用户
⑨1 + 9	传送数据长度	设定待写入的数据块数目	1 到 480	用户
⑨1 + 10	虚拟	不使用	—	—
⑨1 + 11	虚拟	不使用	—	—
⑨1 + 12	虚拟	不使用	—	—
⑨1 + 13				
⑨1 + 14				
⑨1 + 15				
⑨1 + 16	检测出出错的网络号	固定为 0	—	系统
⑨1 + 17	检测出出错的站号	固定为 0	—	系统

[功能]

[用于 MELSECNET/10, A(1S)J71QE71]

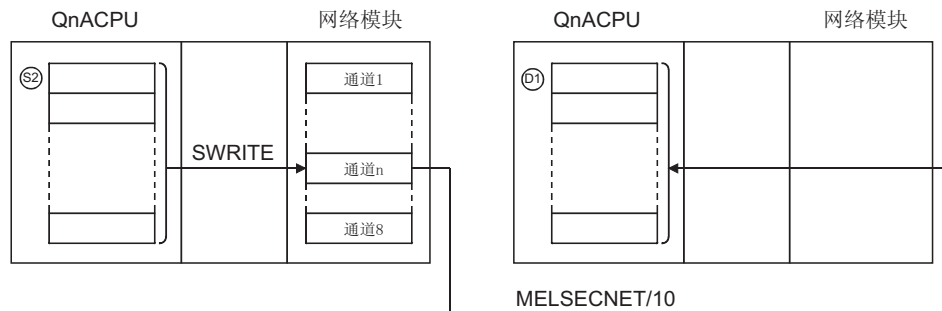
- (1) 将从自站⑥2指定的软元件开始的数据存储到与由控制数据网络号和站号指定的 MELSECNET/10 网络连接的站中从⑥3指定的字数据开始的字软元件中去。

当目标站写入操作已经完成时，被⑥3指定的结束软元件将会变为 ON。

同时，当在目标站中被⑥2指定的软元件数据的写入操作已经完成时，被⑥2指定的软元件将会变为 ON。

[主机站]

[目标站]



- (2) 除了连接到自站网络的站之外，软元件数据的写入操作能够在连接到 MELSECNET/10 指定的网络号的站中执行

- (3) 数据链接指令不能在同一通道的一个以上的地址执行。

当在两个或更多位置同时执行的条件已经建立时，建立收发关系（握手）将被自动执行，所以后续操作中不可能有数据链接指令

- (4) SWRITE 指令是在后台执行，该指令是否正常或出错结束能够通过在使用通道的通讯指示标志 * (自站和目标站中各自的结束软元件 (⑥2和⑥3) 和在自站中的结束软元件 (⑥2+1) 的状态显示) 来确认。

(a) 通讯指示标志

: 在 SWRITE 指令执行时变为 ON，当写入操作已经结束，在扫描周期的 END 处理时变为 OFF。

(b) 自站结束软元件

: 当 SWRITE 指令已经结束时，在扫描周期的 END 处理时变为 ON，在下一个 END 处理时变为 OFF。

(c) 结束时的状态显示软元件

: 变为 ON 和 OFF 取决于 SWRITE 指令结束时的状态

- 正常结束 : 正常结束
- 出错结束 : 当 WRITE 指令已经结束时在扫描周期的 END 处理时变为 ON，在下一个 END 处理时变为 OFF。

[关于 CC-Link, A(1S)J71QC24]

CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

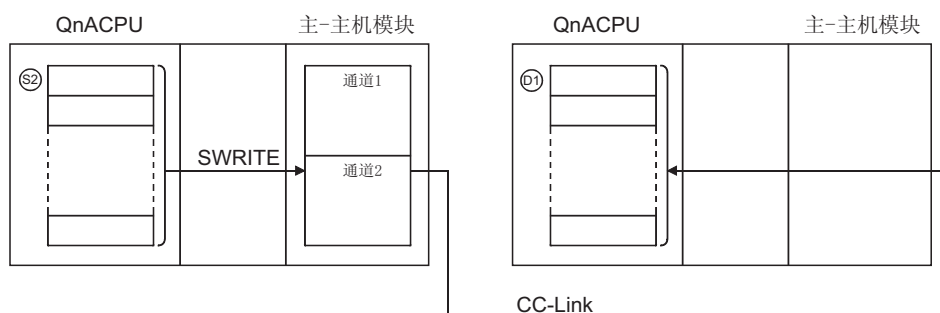
- (1) 将从自站 $\textcircled{S2}$ 指定的软元件开始的数据存储到与由控制数据站号指定的主/自站连接的站中从 $\textcircled{D1}$ 指定的软元件开始的字软元件中去。

当目标站中写入操作已经完成时, 被 $\textcircled{D2}$ 指定的结束软元件将会变为 ON。

同时, 当在目标站中由 $\textcircled{S2}$ 指定的软元件数据的写入操作已经完成时, 被 $\textcircled{D3}$ 指定的软元件将会变为 ON。

[主机站]

[目标站]



- (2) 数据写入操作可以在与 CC-Link 连接 的站上执行。

- (3) 数据链接指令不能在同一通道的多个位置执行。

当在两个或更多位置同时执行的条件已经建立时, 建立收发关系(握手)将被自动执行, 所以后续处理中将不能执行数据链接指令。

- (4) SWRITE 指令是否在执行中, 该指令是正常或错误结束, 能够通过使用通道的通讯指示标志*, 自站和目标站各自的结束软元件($\textcircled{D2}$ 和 $\textcircled{D3}$)和自站的结束软元件($\textcircled{D2}+1$)的状态显示来确认。

(a) 通讯指示标志

: 在 SWRITE 指令执行时变为 ON, 并且当写入操作已经结束, 在扫描周期的 END 处理时变为 OFF。

(b) 自站结束软元件

: 当 SWRITE 指令已经结束时, 在扫描周期的 END 处理时变为 ON, 在下一个 END 处理时变为 OFF。

(c) 结束时的状态显示软元件

: 变为 ON 和 OFF 取决于 SWRITE 指令结束时的状态

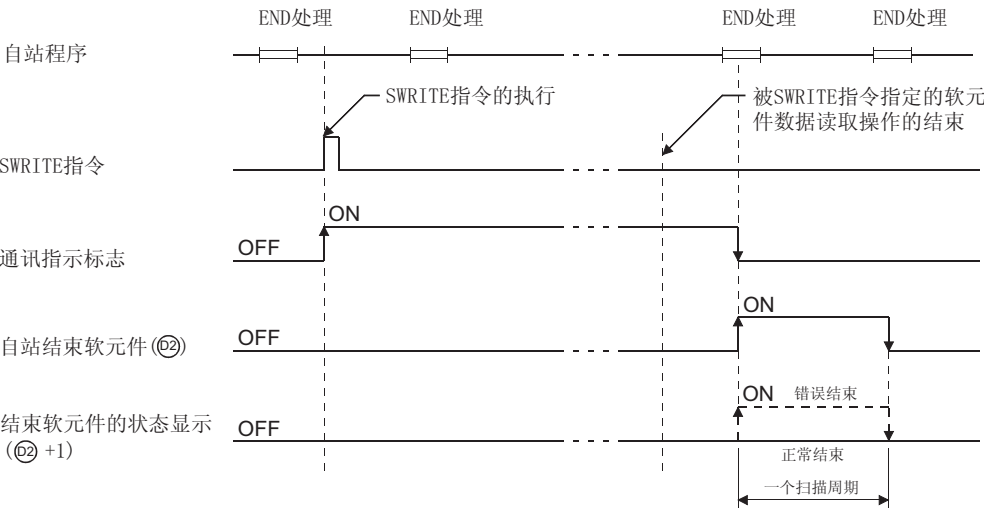
- 正常结束 : 出错结束
- 出错结束 : 当 WRITE 指令已经结束时在扫描周期的 END 处理时变为 ON, 在下一个 EN 处理时变为 OFF。

备注

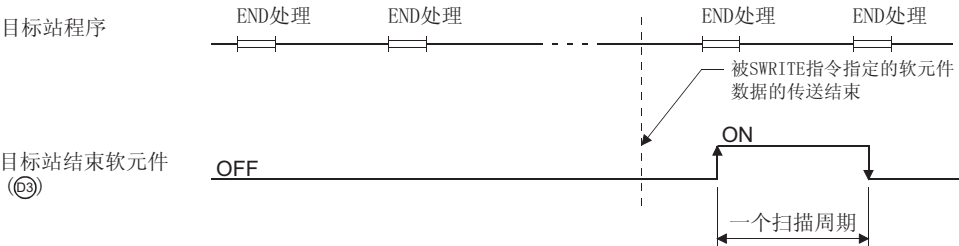
1) *：使用的通道的通讯指示标志如下表示，用于 CC-Link 和 A(1S)J71QC24，最多有两个通道。

通道号	1	2	3	4	5	6	7	8
通讯指示标志号	SB30	SB32	SB34	SB36	SB38	SB3A	SB3C	SB3E

[当 SWRITE 指令被执行时的自站操作]



[当 SWRIE 指令被执行时目标站的操作]



[运行错误]

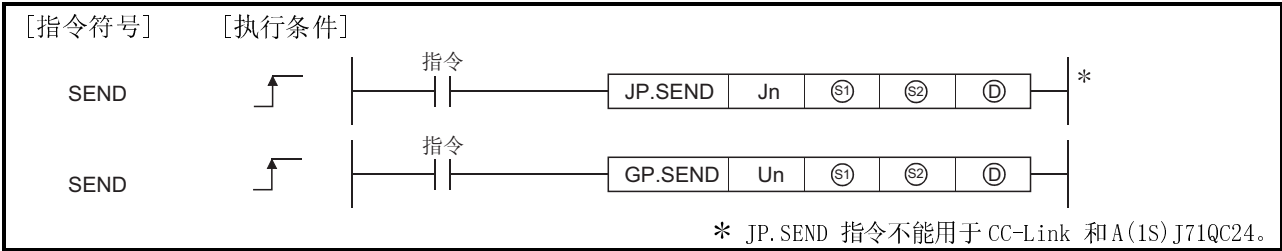
(1) 在以下发生运行错误情况中，错误标志(SM0)变为 ON，并且错误代码存储在 SD0。

- 控制数据不在设定范围内(错误代码：4100)。
- 被 Jn 指定的网络号没有连接到自站(错误代码：4102)。
- 被 Un 指定的 I/O 号的模块不是网络模块(错误代码：2111)。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×*4	×*4	×*4	○	○

8.2.5 传送数据到其它站(SEND)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[]\G[]		特殊功能模块 U[]\G[]	变址寄存器 Zn	常数	其它
	位	字		位	字				
①	—	○	—						
②	—	○	—						
③	○	○	—						



[设定数据]

[用于 MELSECNET/10, A(1S)J71QE71]

设定数据	含义	数据类型
Jn	自站网络号 *1	BIN 16 位
Un	自站链接模块的起始 I/O 号 *2	
①	存储控制数据的软元件的起始软元件号	软元件名称
②	存储将要传送的数据的软元件的起始软元件号	
③	在传送结束时, 变为 ON 一个扫描周期的软元件 *3	位

备注

- *1: 自站网络号按如下指定:
- 1 到 239 : 网络号
 - 254 : 由设定为其它站可以访问的有效模块来指定的网络
- *2: 自站网络模块的起始 I/O 号的指定可以为 0 到 FEH。
- *3: 每一个程序的局部软元件和文件寄存器均不能当作设定数据使用的软元件来使用。(详细资料参见 QnACPU 编程手册(特殊功能模块))
- *4: 能够在 QCPU 中使用的 SEND 指令的详细资料, 参见以下手册:
- Q 系列 MELSECNET/H 网络系统参考手册(PLC 到 PLC 网络)
 - Q 系列以太网接口模块用户手册(应用篇)

[关于 CC-Link, A(1S)J71QC24]

CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

设定数据	含义	数据类型
Un	自站链接模块的起始 I/O 号 *1	BIN 16 位
①	存储控制数据的软元件的起始软元件号	软元件名称
②	存储待传送的数据的软元件的起始软元件号	
③	在传送结束时在一个扫描周期内变为 ON 的软元件 *2	位

备注

- *1: 自站主/本模块的起始 I/O 号的可以指定为 0 到 FEH。
- *2: 每一个程序的局部软元件和文件寄存器均不能当作设定数据使用的软元件。
(详细资料参见 QnACPU 编程手册(特殊功能模块))

[控制数据]

[用于 MELSECNET/10, A(1S)J71QE71]

软元件	项目	设定数据	设定范围	设定者
Ⓔ + 0	执行类型	无传送结束确认 : 设定位 0 到 0 传送结束确认 : 设定位 0 到 0	0000 _h	用户
	出错结束类型	当出错处理结束时设定时钟数据设定状态 不设定时钟数据 : 设定位 7 (b7) 为 0 设定时钟数据 : 设定位 7 (b7) 为 1 (Ⓔ+10 向前附加)	0001 _h 0080 _h 0081 _h	
Ⓔ + 1	结束状态	在指令结束时的状态被存储 0 : 无出错 (正常结束) 除 0 之外 : 错误代码 * 1	—	系统
Ⓔ + 2	被自站使用的通道	指定自站将要使用的通道 * 2	1 到 8	用户
Ⓔ + 3	目标站使用的通道	为将要存储数据的目标站指定通道	1 到 8	用户
Ⓔ + 4	目标站的网络号	为目标站指定网络号	1 到 239 254 * 3	用户
Ⓔ + 5	目标站号	指定目标站的站号 (仅当执行类型为 0 时设定范围可以为 81 _h 到 89 _h)	1 到 64: 站号 81 _h 到 89 _h : 组指定 FF _h : 在目标网络中的 所有站	用户
Ⓔ + 6	虚拟	不使用	—	—
Ⓔ + 7	传送重试次数	当执行类型设定为 1 时有效, 当 WDT 被 Ⓔ+8 指定的时间内没有完成结束时, 设定传送重试次数。	1 到 15	用户
	重试结果	存储重复处理的次数		系统
Ⓔ + 8	到达的 WDT 时间	指令结束的 WDT 时间是以秒为单位设定的。如果在 WDT 设定的时间内操作还没有结束时, 按照被 Ⓔ+7 指定的次数传送被重试。	1 到 32767 0 : 10 秒固定 仅当执行类型在 1 处设定时有效。	用户
Ⓔ + 9	传送数据长度	设定将要传送的数据数目	1 到 480	用户
Ⓔ + 10	时钟设定标志 (只在出错时设定)	存储时钟数据的允许禁止状态 当时钟数据禁止 : 0 当时钟数据禁止 : 1	—	系统
Ⓔ + 11	时钟数据 (只在出错时设定)	高 8 位 : 月 (1 到 12), 低 8 位 : 年 (0 到 99)	—	系统
Ⓔ + 12		高 8 位 : 时 (0 到 23), 低 8 位 : 日 (1 到 31)		
Ⓔ + 13		高 8 位 : 秒 (0 到 59), 低 8 位 : 分 (0 到 59)		
Ⓔ + 14		高 8 位 : 00 _h , 低 8 位 : 星期几 (0 到 6)		
Ⓔ + 16	检测到错误的网络号	存储检测到错误的站的网络号。 1 到 239 : 网络号	—	系统
Ⓔ + 17	检测到错误的站号	存储检测到错误的站的站号。 如果 Ⓔ+1 结束状态为“通道使用中”时, 不存储数据。 1 到 64 : 站号	—	系统

备注

*1: 关于错误代码, 请参考下列手册:

- 错误代码: F000_H 到 FFFF_H
QnA/Q4AR 用 MELSECNET/10 网络系统参考手册
- 错误代码: 4000_H 到 4FFF_H
用户手册中一般数据处理中的返回到要求源的错误代码。
- 以上显示之外的错误代码:
在这种情况下, 错误将是由目标站造成的。
参见 Q2A (S1)/Q3A/Q4ACPU 用户手册中的故障排除。

*2: 网络模块的通道信息, 参见第 8 章。

*3: 当值 254 已经被 Jn 设定时指定。

[关于 CC-Link, A(1S)J71QC24]

CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

软元件	项目	设定内容	设定范围	设定者
⑨1 + 0	执行类型	0: 无到达确认 1: (执行到达确认)	0, 1	用户
⑨1 + 1	结束状态	指令结束时状态被存储 0 : 无错误(正常结束) 除 0 之外 : 错误代码	—	系统
⑨1 + 2	自站使用通道	指定自站将要使用的通道 1: 使用 CH1 通道 2: 使用 CH2 通道	1, 2	用户
⑨1 + 3	目标站使用通道	指定目标站将存储数据的通道 1: 使用 CH1 通道 2: 使用 CH2 通道	1, 2	用户
⑨1 + 4	目标站的网络号	固定为 0	0	用户
⑨1 + 5	目标站号	固定为 0	0	用户
⑨1 + 6	主/自站号	0: 指定的主站 1 到 64: 指定的自站	0 到 64	用户
⑨1 + 7	传送重试次数	当被⑨1指定的执行类型为 1 时有效, 并当 WDT 中在被⑨1+8 指定的时间内没有完成结束, 设定传送重试次数	0 到 15	用户
	重试结果	存储重复处理的次数。	—	系统
⑨1 + 8	到达的 WDT 时间	指令结束的 WDT 时间是以秒为单位设定的。如果操作在为 WDT 设定的时间内还没有结束, 那么传送按照被⑨1+7 指定的次数被重复	1 到 32767 0: 10 秒固定 (缺省) 仅当执行类型在 1 设定时有效。	用户
⑨1 + 9	传送数据长度	设定要传送数据的号	1 到 480	用户
⑨1 + 10	虚拟	不使用	—	—
⑨1 + 11	虚拟	不使用	—	—
⑨1 + 12				
⑨1 + 13				
⑨1 + 14				
⑨1 + 16	检测到错误的网络号	固定为 0	—	系统
⑨1 + 17	检测到错误的站号	固定为 0	—	系统

[功能]

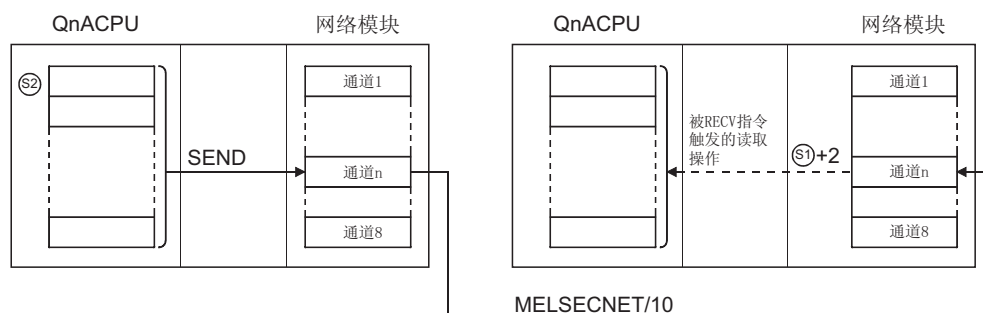
[用于 MELSECNET/10, A(1S)J71QE71]

- (1) 从自站⑤2指定的软元件开始的数据存储到与由控制数据网络号和站号指定的 MELSECNET/10 网络模块连接的站的指定通道中去。

当目标站中写入操作已经完成时，被⑤指定的结束软元件将会变为 ON

[自站]

[目标站]



- (2) 除了连接到自站网络的站之外，软元件数据的写入操作能够在连接到 MELSECNET/10 指定的网络号的站中执行。

- (3) 在同一个通道中，数据链接指令不能在多个位置上执行。

当在两个或更多位置同时执行的条件已经建立时，建立收发关系（握手）将被自动执行，所以后续处理中将不能执行数据链接指令。

- (4) SEND 指令是否在执行中，该指令是正常或错误结束，能够通过使用通道的通讯指示标志*，结束软元件⑤和在自站的结束状态显示软元件(⑤+1)来确认。

(a) 通讯指示标志

: 在 SEND 指令执行时变为 ON，当写入操作已经结束，在扫描周期的 END 处理时变为 OFF。

(b) 自站结束软元件

: 当被 SEND 指令触发的写入操作已经结束时，在扫描周期的 END 处理时变为 ON，在下一个 END 处理时变为 OFF。

(c) 结束时的状态显示软元件

: 变为 ON 和 OFF 取决于 SEND 指令结束时的状态

- 正常结束 : 保持 OFF，不改变
- 出错结束 : 当 SEND 指令结束时在扫描周期的 END 处理时变为 ON，在下一个 END 处理时变为 OFF。

[关于 CC-Link, A(1S)J71QC24]

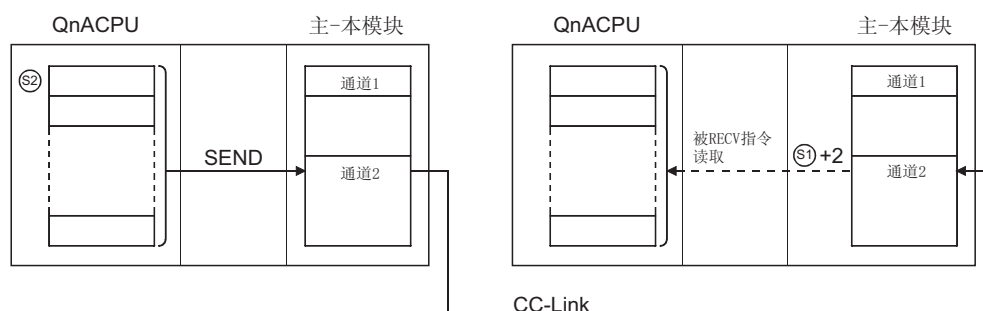
CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

- (1) 将从自站⑤指定的软元件开始的数据存储到与由控制数据站号指定的主/本模块连接的站中指定的通道中去。

当目标站中写入操作已经完成时, 被⑤指定的结束软元件将会变为 ON

[自站]

[目标站]



- (2) 软元件数据的写入操作能够在连接到 CC-Link 的站中执行。

- (3) 在同一个通道中, 数据链接指令不能在多个位置上执行。

当在两个或更多位置同时执行的条件已经建立时, 建立收发关系(握手)将被自动执行, 所以后续处理中将不能执行数据链接指令。

- (4) SEND 指令是否在执行中, 该指令是正常或错误结束, 能够通过使用通道的通讯指示标志*, 结束软元件⑤, 和自站中的结束状态显示软元件(⑤+1)来确认。

(a) 通讯指示标志

: 在 SEND 指令执行时变为 ON, 当写入操作结束后, 在扫描周期的 END 处理时变为 OFF。

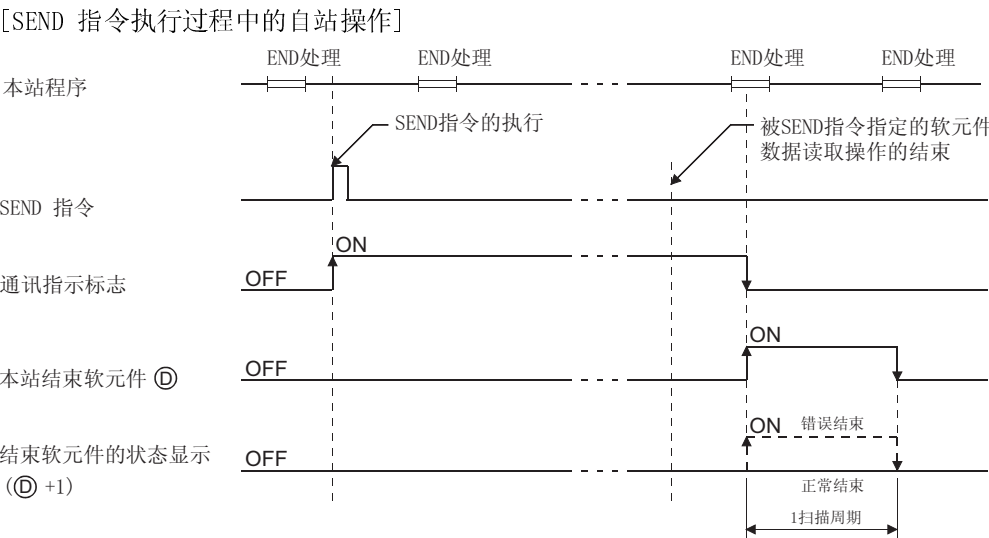
(b) 自站结束软元件

: 当被 SEND 指令触发的写入操作已经结束时, 在扫描周期的 END 处理时变为 ON, 在下一个 END 处理时变为 OFF。

(c) 结束时的状态显示软元件

: 变为 ON 和 OFF 取决于 SEND 指令结束时的状态

- 正常结束 : 出错结束
- 出错结束 : 当 SEND 指令结束时在扫描周期的 END 处理时变为 ON, 在下一个 END 处理时变为 OFF。



- [运行错误]
- (1) 在以下发生运行错误情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0。
- 控制数据的内容不在设定范围内 (错误代码: 4100)
 - 被 Jn 指定的网络号没有连接到这个站 (错误代码: 4102)
 - 被 Un 指定 I/O 号的模块不是网络模块 (错误代码: 2111)

备注

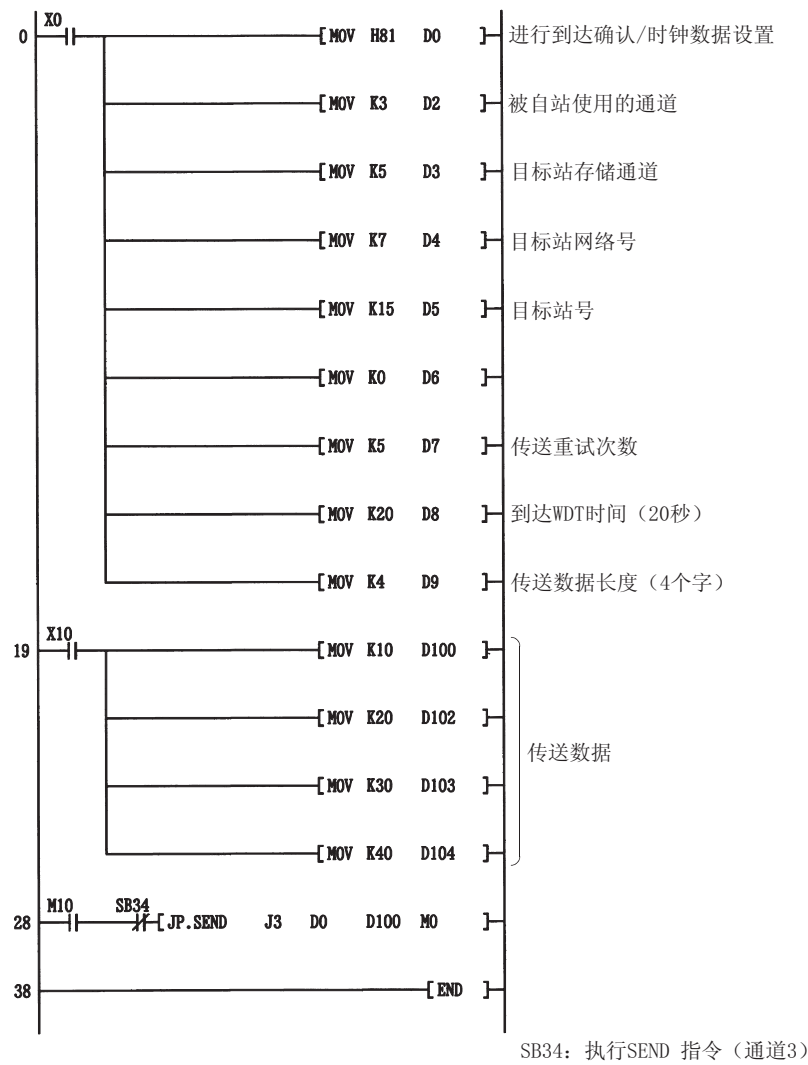
- 1) *:被使用通道的通讯指示标志列示如下:
用于 CC-Link 和 A(1S) J71QC24, 最多有两个通道可以使用。

通道号	1	2	3	4	5	6	7	8
通讯指示标志号	SB30	SB32	SB34	SB36	SB38	SB3A	SB3C	SB3E

[程序示例]

(1) 下列程序通过 SEND 指令使用 7 号网络上的网络模块中的通道 3 来传送数据到 5 号网络上的第 15 号站的通道 5 中去。

[梯形图模式]





[列表模式]

步	指令	软元件
0	LD	X0
1	MOV	H81
		D0
3	MOV	K3
		D2
5	MOV	K5
		D3
7	MOV	K7
		D4
9	MOV	K15
		D5
11	MOV	K0
		D6
13	MOV	K5
		D7
15	MOV	K20
		D8
17	MOV	K4
		D9
19	LD	X10
20	MOV	K10
		D100
22	MOV	K20
		D102
24	MOV	K30
		D103
26	MOV	K40
		D104
28	LD	M10
29	ANI	S834
30	JP.SEND	J3
		D0
		D100
		M0
38	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×*5	×*5	×*5	○	○

8.2.6 从其他站点接收数据 (RECV)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[]\G[]		特殊功能模块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○					—
①①	—	○					—
②②	○	○					—

[指令符号]	[执行条件]
RECV	
RECV	
* JP.RECV 指令不能用于 CC-Link 和 A(1S)J71QC24。	

[设定数据]

[用于 MELSECNET/10, A(1S)J71QE71]

设定数据	含义	数据类型
Jn	本站的网络号 *1	BIN 16 位
Un	本站链接模块的起始 I/O 号 *2	
⑤	存储控制数据的起始软元件号	软元件名称
①①*4	存储传送数据的起始软元件号	
②②	在传送结束后在一个扫描周期内变为 ON 的软元件*3	位

备注

- *1: 本站网络号按如下指定:
- 1 到 239 : 网络号
 - 254 : 由设定为其它站可以访问的有效模块来指定的网络
- *2: 本站网络模块的起始 I/O 号的指定可以为 0 到 FE₁₆。
- *3: 每个程序的局部软元件和文件寄存器均不能当作为设定数据使用的软元件来使用 (详细情况请参见 QnACPU 编程手册 (特殊功能模块)。)
- *4: 当接收数据软元件范围超过指定的限制, 那么 4031H (软元件范围上限) 将在结束状态中设定。
- *5: 可以在 QCPU 中使用的 WRITE 指令的详细情况, 请参见下列手册:
- Q 系列 MELSECNET/H 网络系统参考手册 (PLC 到 PLC 网络)
 - Q 系列以太网 5 接口模块用户手册 (应用篇) *1。

[关于 CC-Link, A(1S)J71QC24]

CC-Link 可以用软件版本 J 或更新版本的主模块 (A(1S)J61QTB11)

设定数据	含义	数据类型
Un	本站链接模块的起始 I/O 号*1	BIN 16 位
⑤	存储控制数据的软元件的起始软元件号	软元件名称
①①*3	存储传送数据的软元件的起始软元件号	
②②	在传送结束时, 将在一个扫描周期内变为 ON 的软元件*2	位

备注

- *1: 自站主/本模块的起始 I/O 号的指定可以为 0 到 FEH。
 *2: 每一个程序的局部软元件和文件寄存器均不能当作设定数据使用的软元件来使用。
 (详细情况请参考 QnACPU 编程手册 (特殊功能模块)。)
 *3: 当接收数据软元件的范围超过指定的限制时, 那么 4031H (软元件范围上限) 将在结束状态中设定。

[控制数据]

用于 MELSECNET/10, A(1S)J71QE71

软元件	项目	设定数据	设定范围	设定者
S1 + 0	执行类型	当无数据时等待 (在数据接收到之前等待固定周期时间) (位 0 = 0 (固定))	0000H 0080H	用户
	错误结束类型	管理时钟数据设定在出错结束时允许禁止 不设定时钟数据 : 设定位 7 (b7) 到 0 设定时钟数据 : 设定位 7 (b7) 到 1 (S1+11 向前附加)		
S1 + 1	结束状态	指令结束时的状态被存储。 0 : 无错误 (正常结束) 除 0 之外 : 错误代码 *1	—	系统
S1 + 2	自站使用通道	指定读取数据正在被存储的通道 *2	1 到 8	用户
S1 + 3	目标站使用通道	存储被传送站使用的通道	1 到 8	系统
S1 + 4	目标站的网络号	存储传送站的网络号	1 到 239	系统
S1 + 5	目标站站号	存储传送站的站号	1 到 64	系统
S1 + 6	虚拟	不使用	—	—
S1 + 7	虚拟	不使用	—	—
S1 + 8	到达 WDT 时间	指令结束的 WDT 时间是以秒为单位设定的。如果在 WDT 时间内接收还没有结束将会出现错误结束	1 到 32767 0 : 10 秒固定 仅当执行类型在 1 处设定时有效。	用户
S1 + 9	接收允许的数据长度	存储接收到的数据块数目	1 到 480	系统
S1 + 10	虚拟	不使用	—	—
S1 + 11	时钟设定标志 (只在出错时设定)	存储时钟数据的允许禁止状态 当时钟数据禁止 : 0 当时钟数据允许 : 1	—	系统
S1 + 12	时钟数据 (只在出错时设定)	高 8 位 : 月 (1 到 12), 低 8 位 : 年 (0 到 99)	—	系统
S1 + 13		高 8 位 : 时 (0 到 23), 低 8 位 : 日 (1 到 31)		
S1 + 14		高 8 位 : 秒 (0 到 59), 低 8 位 : 分 (0 到 59)		
S1 + 15		高 8 位 : 00H, 低 8 位 : 星期 (0 到 6)		

备注

- *1: 关于错误代码, 请参考下列手册:
 QnA/Q4AR MELSECNET/10 网络系统参考手册
 *2: 网络模块通道的信息, 请参见第 8 章。

[关于 CC-Link, A(1S)J71QC24]

CC-Link 可以用软件版本 J 或更新版本的主模块(A(1S)J61QTB11)

软元件	项目	设定数据	设定范围	设定者
⑨ + 0	执行类型	固定为 0 (无到达确认)	0	用户
⑨ + 1	结束状态	在指令结束时的状态被存储 0 : 无出错 (正常结束) 除 0 之外 : 错误代码	—	系统
⑨ + 2	自站使用通道	指定读取数据正在存储的通道。 1 : 使用 CH1 通道 2 : 使用 CH2 通道	1, 2	用户
⑨ + 3	目标站使用通道	存储被传送站使用的通道 1 : 使用 CH1 通道 2 : 使用 CH2 通道	1, 2	系统
⑨ + 4	目标站的网络号	固定为 0	0	系统
⑨ + 5	目标站号	固定为 0	0	系统
⑨ + 6	主/自站号	0: 指定的主站 1 到 64: 指定的自站	0 到 64	用户
⑨ + 7	虚拟	不使用	—	—
⑨ + 8	虚拟	不使用	—	—
⑨ + 9	接收允许的数据长度	存储接收到数据块数目	1 到 480	系统
⑨ + 10	虚拟	不使用	—	—
⑨ + 11	虚拟	不使用	—	—
⑨ + 12				
⑨ + 13				
⑨ + 14				
⑨ + 15				

[功能]

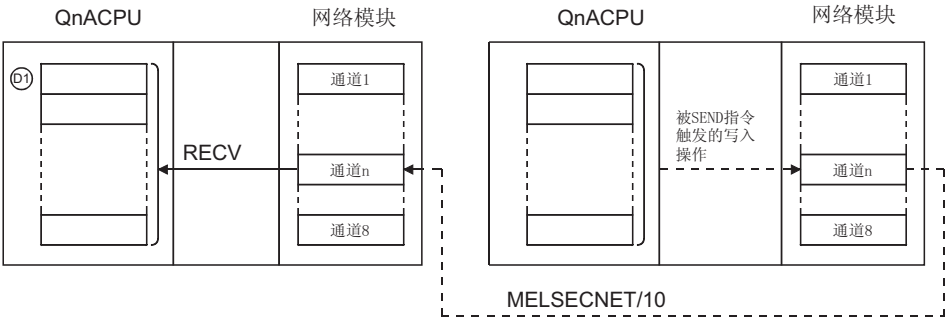
用于 MELSECNET/10, A(1S)J71QE71

(1) 在自站的网络模块中被⑨指定的软元件之后, 从连接到被控制数据网络号和站号指定的 MELSECNET/10 网络的站中, 存储通过 SEND 指令传送的数据。

当从网络模块中读取软元件数据的操作已经完成, 被⑨指定的结束软元件将会变为 ON。

[自站]

[目标站]



(2) 在同一个通道中, 数据链接指令不能在多个位置上执行。

当在两个或更多位置同时执行的条件已经建立时, 建立收发关系 (握手) 将被自动执行, 所以后续处理中将不能执行数据链接指令。

(3) RECV 指令是否在执行中，该指令是正常或错误结束能够通过使用通道的通讯指示标志*，结束软元件②，和在自站的结束软元件的状态显示 (②+1) 来确认。

(a) 通讯指示标志

: 在 RECV 指令执行时变为 ON，当写入操作结束后，在扫描周期的 END 处理时变为 OFF。

(b) 自站结束软元件

: 当被 RECV 指令触发的写入操作已经结束时，在扫描周期的 END 处理时变为 ON，在下一个 END 处理时变为 OFF。

(c) 结束时的状态显示软元件

: 变为 ON 或 OFF 取决于 RECV 指令结束时的状态

- 正常结束 : 保持 OFF 不变
- 出错结束 : 当 RECV 指令已经结束时在扫描周期的 END 处理时变为 ON，在下一个 END 处理时变为 OFF。

[关于 CC-Link, A(1S)J71QC24]

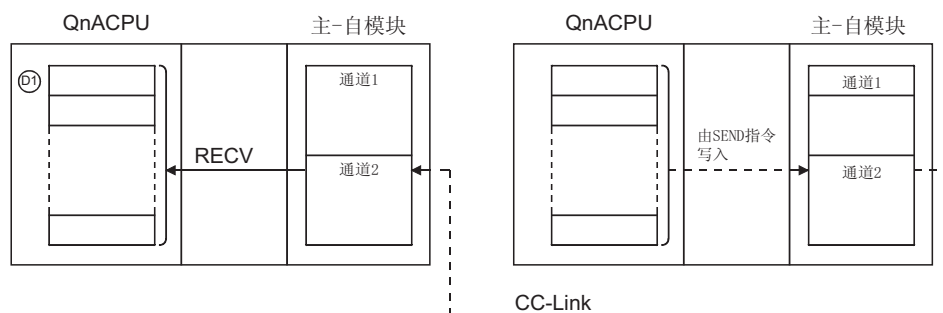
CC-Link 可以用软件版本 J 或更新的主模块(A(1S)J61QTB11)

(1) 在自站的网络模块中被②指定的软元件之后，从连接到被控制数据站号指定站号的主/自站的站中，存储通过 SEND 指令传送的数据。

当从网络模块中读取软元件数据的操作已经完成，被②指定的结束软元件将会变为 ON。

[自站]

[目标站]



(2) 在同一个通道中，数据链接指令不能在多个位置上执行。

当在两个或更多位置同时执行的条件已经建立时，建立收发关系（握手）将被自动执行，所以后续处理中将不能执行数据链接指令。

(3) RECV 指令是否在执行中，该指令是否正常或出错结束能够通过在使用通道的通讯指示标志，结束软元件②，和在自站中的结束软元件的状态显示(②+1)来确认。

(a) 通讯指示标志

：在 RECV 指令执行时变为 ON，当写入操作结束后，在扫描周期的 END 处理时变为 OFF。

(b) 自站结束软元件

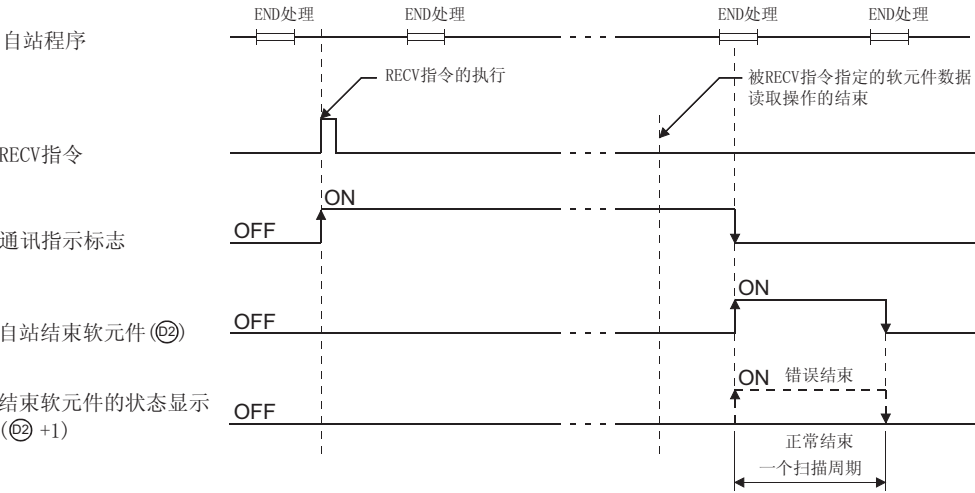
：当被 RECV 指令触发的写入操作已经结束时，在扫描周期的 END 处理时变为 ON，在下一个 END 处理时变为 OFF。

(c) 结束时的状态显示软元件

：变为 ON 和 OFF 取决于 RECV 指令结束时的状态。

- 正常结束 ：保持 OFF 不变
- 出错结束 ：当 RECV 指令已经结束时在扫描周期的 END 处理时变为 ON，在下一个 END 处理时变为 OFF。

[RECV 指令被执行时的自站操作]



[运行错误]

- (1) 在以下发生运行错误情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0。
- 控制数据的内容不在设定范围内 (错误代码：4100)。
 - 被 Jn 指定的网络号没有连接到这个站 (错误代码：4102)。
 - 被 Un 指定 I/O 号的模块不是网络模块 (错误代码：2111)。

备注

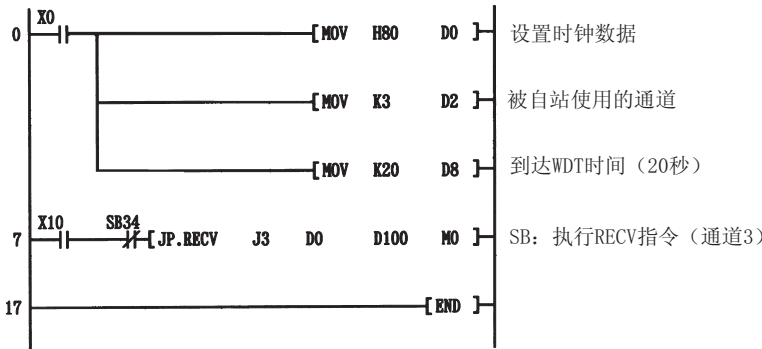
- 1) *：被使用通道的通讯指示标志如下所示：
对于 CC-Link 和 A(1S) J71QC24, 最多有两个通道可以使用。

通道号	1	2	3	4	5	6	7	8
通讯指示标志号	SB30	SB32	SB34	SB36	SB38	SB3A	SB3C	SB3E

[程序示例]

- (1) 下述程序从第 3 网络的网络模块中的通道 3 读取通过 SEND 指令传送的数据。

[梯形图模式]



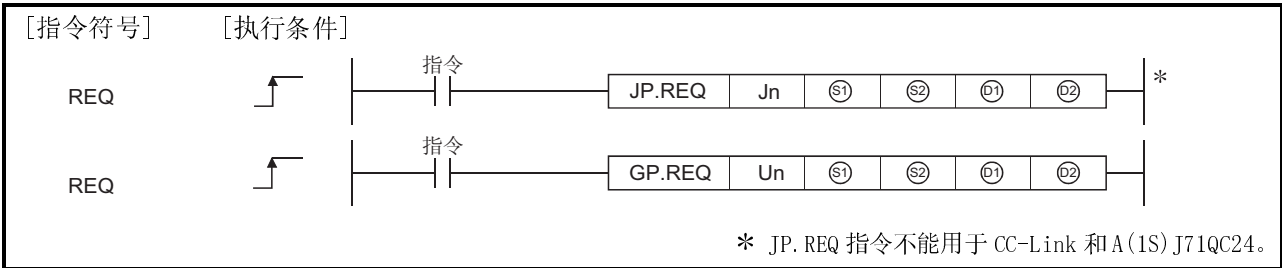
[列表模式]

步	指令	软元件
0	LD	X0
1	MOV	H80 D0
3	MOV	K3 D2
5	MOV	K20 D8
7	LD	X10
8	ANI	SB34
9	JP.RECV	J3 D0 D100 M0
17	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	×	×	○	○

8.2.7 来自其它站的瞬时请求(读/写数据块，远程 RUN/STOP) (REQ)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[]		特殊功能模块 U[]G[]	变址寄存器 Zn	常数	其它
	位	字		位	字				
S1	—	○				—			
S2	—	○				—			
D1	—	○				—			
D2	○	○				—			



[设定数据]

[对于 MELSECNET/10, A(1S)J71QE71]

设定数据	含义	数据类型
Jn	本站的网络号 *1	BIN 16 位
Un	本站链接模块的起始输入输出 (I/O) 号 *2	
S1	存储控制数据的软元件的起始软元件号	软元件名称
S2	存储请求数据的软元件的起始软元件号	
D1	存储响应数据的软元件的起始软元件号	
D2	在指令结束时，软元件变为 ON 一个扫描周期 *3	位

备注

- *1: 本站的网络号指定如下：
 - 1 到 239 : 网络编号
 - 254 : 由来自其它站的设定为访问有效模块所指定的网络。
- *2: 本站主/自模块的起始输入输出 (I/O) 号可指定为从 0 到 FE_H。
- *3: 每一个程序的局部软元件和文件寄存器不能被用作一个用于设定数据的软元件（具体细节请参照 QnACPU 编程手册（特殊功能模块）。）

[关于 CC-Link, A(1S)J71QC24]

CC-Link 可以用软件版本 J 或更新版本的主模块 (A(1S)J61QTB11)

设定数据	含义	数据类型
Un	位于链接模块上的本站起始输入输出 (I/O) 号 *1	BIN 16 位
S1	存储控制数据的软元件的起始软元件号	软元件名称
S2	存储请求数据的软元件的起始软元件号	
D1	存储响应数据的软元件的起始软元件号	
D2	指令结束时，软元件变为 ON 一个扫描周期 *2	位

备注

- *1: 本站主/自模块的起始输入输出 (I/O) 号可指定为从 0 到 FE_H。
- *2: 每一个程序的局部软元件和文件寄存器不能被用作一个用于设定数据的软元件（具体细节请参照 QnACPU 编程手册(特殊功能模块)。）

要点
(1) 仅当目标站为一个 QCPU/QnACPU 站时，REQ 指令可以被执行。 (在一个连接到 MELSECNET/10 网络的 ACPU 站中不能执行 REQ 指令。) 对于 QCPU/QnACPU 只有一个站号将被设定成目标站号。

[控制数据]

对于 MELSECNET/10, A(1S)J71QE71

软元件	条目	设定数据	设定范围	设定者
	执行类型	确认已到达 (位 0=1 (固定的))		
Ⓔ + 0	错误的结束类型	当出错处理结束时设定时钟数据设定状态。 不设定时钟数据 : 设定位 7 (b7) 到 0 设定时钟数据 : 设定位 7 (b7) 到 1 (Ⓔ + 11 向前附加)	001 _{1h} 009 _{1h}	用户
Ⓔ + 1	结束状态	存储指令的结束状态 0 : 没有出错 (正常结束) 其它不为 0 的 : 错误代码 * 1	—	系统
Ⓔ + 2	自站使用的通道	自站将使用的指定通道 * 2	1 到 8	用户
Ⓔ + 3	虚拟	未使用	—	—
Ⓔ + 4	目标站网络号	目标站指定的网络号	1 到 239 254 * 3	用户
Ⓔ + 5	目标站站号	目标站指定的站号	1 到 64	用户
Ⓔ + 6	虚拟	未使用	—	—
Ⓔ + 7	发送重试次数	如果发送在由Ⓔ+8 指定的 WDT 时间内没有完成, 设定的重试的次数。	1 到 15	用户
	重试结果	存储重复处理的次数。	—	系统
Ⓔ + 8	到达 WDT 时间	指令结束的 WDT 时间以秒为单位。 如果在设定时间内读操作没有完成, 将会按由Ⓔ+7 指定的次数重新发送。	1 到 32767 0 : 10 秒固定的	用户
Ⓔ + 9	请求数据长度	指定请求数据长度 正在读时钟数据时 : 2 正在写时钟数据时 : 6 在远程 RUN/STOP 期内 : 4	2, 6, 4	用户
Ⓔ + 10	响应数据长度	存储响应数据长度 在远程 RUN/STOP 期内 : 2 正在写时钟数据时 : 2 正在写时钟数据时 : 6	2, 6	系统
Ⓔ + 11	时钟设定标记 (仅当出错时设定)	存储时钟数据许可/禁止状态 存储时钟数据许可/禁止状态 : 0 当时钟数据为许可状态时 : 1	—	系统
Ⓔ + 12	时钟数据 (仅当出错时设定)	高 8 位: 月 (1 到 12) 低 8 位: 年 (0 到 99)	—	系统
Ⓔ + 13		高 8 位: 小时 (0 到 23) 低 8 位: 日 (每月) (1 到 31)		
Ⓔ + 14		高 8 位: 秒 (0 到 59) 低 8 位: 分钟 (0 到 59)		
Ⓔ + 15		高 8 位: 00 _h 低 8 位: 星期 (0 到 6)		
Ⓔ + 16	检测到错误的网络号	如果结束状态(Ⓔ+1) 为 “通道正在使用 (F7C1 _h)”, 那么检测到错误的站的网络号将不会被存储。 1 到 239 : 网络号	—	系统
Ⓔ + 17	检测到错误的站号	如果结束状态(Ⓔ+1) 为 “通道正在使用 (F7C1 _h)”, 那么检测到错误的站的站号将不会被存储。 1 到 64 : 站号	—	系统

备注

- *1: 相关错误代码信息请参照以下手册：
QnA/Q4AR MELSECNET/10 网络系统参考手册
- *2: 网络模块通道的有关信息请参见第 8 章。
- *3: 当值 254 由 Jn 设定时指定。

关于 CC-Link, A(1S)J71QC24

CC-Link 可以用软件版本 J 或更新的主模块 (A(1S)J61QTB11)

软元件	条目	设定数据	设定范围	设定者
⑨1 + 0	执行类型	确认已到达 (位 0=1 (固定的))	1	用户
⑨1 + 1	结束状态	存储指令的结束状态 0 : 无错误 (正常结束) 其它大于 0 的 : 错误代码	—	系统
⑨1 + 2	自站使用的通道	自站将使用的指定通道 1 : 使用 CH1 通道 2 : 使用 CH2 通道	1, 2	用户
⑨1 + 3	虚拟	未使用	—	—
⑨1 + 4	目标站网络号	一定为 0	0	用户
⑨1 + 5	目标站站号	一定为 0	0	用户
⑨1 + 6	主/自站号	0: 指定主站 1 到 64: 指定的自站	0 到 64	用户
⑨1 + 7	发送重试次数 重试结果	如果发送在由⑨1+8 指定的 WDT 时间内没有完成, 设定的重试的次数。 存储重复处理的次数。	0 到 15 —	用户 系统
⑨1 + 8	到达 WDT 时间	指令结束的 WDT 时间以秒为单位。 如果在设定时间内读操作没有完成, 将会按由⑨1+7 指定的次数重新发送。	1 到 32767 0:10 秒固定 (默认)	用户
⑨1 + 9	请求数据长度	指定请求数据长度 在远程 RUN/STOP 期内 : 4	4	—
⑨1 + 10	响应数据长度	存储响应数据长度 在远程 RUN/STOP 期内 : 4	—	系统
⑨1 + 11	虚拟	未使用	—	—
⑨1 + 12	虚拟	未使用	—	—
⑨1 + 13				
⑨1 + 14				
⑨1 + 15				
⑨1 + 16	检测到错误的网络号	固定为 0	—	系统
⑨1 + 17	检测到错误的站号	固定为 0	—	系统

[请求数据/响应数据]

[对于 MELSECNET/10, A(1S)J71QE71]

(1) 在对时钟数据的读/写操作中

① 请求数据

软元件	条目	含义	读时钟数据	写时钟数据
Ⓔ	请求类型	0001 _H : 读时钟数据 0011 _H : 写时钟数据	○	○
Ⓔ + 1	子请求类型	0002 _H : 读时钟数据 0001 _H : 写时钟数据	○	○
Ⓔ + 2	更新格式 被更新的年份	① 更新格式 (位 0 到 7) 指定将被写入的Ⓔ+2 到Ⓔ+5 的增序位的条目 0: 更新 1: 不更新 ② 更新年份 (位 8 到 15) 年 (年份的最后两位) 以 BCD 码形式存储 <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;"> b15 to b8 b7 b6 b5 b4 b3 b2 b1 b0 </div> <div style="border: 1px solid black; padding: 2px;"> 年 (00_H到99_H) 0 </div> </div> <div style="margin-left: 100px;"> 年 月 日 小时 分钟 秒 日期 (一周) </div>		○
Ⓔ + 3	被更新的月份和日	月和日以 BCD 码形式存储 <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">b15 to b8</div> <div style="margin-right: 10px;">b7 to b0</div> </div> <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">日 (01_H到31_H)</div> <div style="border: 1px solid black; padding: 2px;">月 (01_H到12_H)</div> </div>		○
Ⓔ + 4	被更新的小时和分钟	小时和分钟以 BCD 码形式存储 <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">b15 to b8</div> <div style="margin-right: 10px;">b7 to b0</div> </div> <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">分钟 (00_H到59_H)</div> <div style="border: 1px solid black; padding: 2px;">小时 (00_H到23_H)</div> </div>		○
Ⓔ + 5	被更新的秒和日期 (一周)	秒和日期 (一周) 以 BCD 码形式存储 <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">b15 to b8</div> <div style="margin-right: 10px;">b7 to b0</div> </div> <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">日期 (一周) (00_H到06_H)</div> <div style="border: 1px solid black; padding: 2px;">秒 (00_H到59_H)</div> </div> <div style="margin-left: 100px;"> 从 00_H (星期天) 到 06_H (星期六) </div>		○

② 响应数据

软元件	条目	含义	读时钟数据	写时钟数据
Ⓕ	请求类型	0081 _H : 读时钟数据 0091 _H : 写时钟数据	○	○
Ⓕ + 1	子请求类型	0002 _H : 读时钟数据 0001 _H : 写时钟数据	○	○
Ⓕ + 2	被读取的月份和年份	月份和年份 (年份的最后两位) 以 BCD 码形式存储。 <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">b15 to b8</div> <div style="margin-right: 10px;">b7 to b0</div> </div> <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">月 (01_H到12_H)</div> <div style="border: 1px solid black; padding: 2px;">年 (00_H到99_H)</div> </div>	○	
Ⓕ + 3	被读取的小时和日期	小时和天以 BCD 码形式存储。 <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">b15 to b8</div> <div style="margin-right: 10px;">b7 to b0</div> </div> <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">小时 (00_H到23_H)</div> <div style="border: 1px solid black; padding: 2px;">天 (01_H到31_H)</div> </div>	○	
Ⓕ + 4	被读取的秒和分钟	秒和分钟以 BCD 码形式存储。 <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">b15 to b8</div> <div style="margin-right: 10px;">b7 to b0</div> </div> <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">秒 (00_H到59_H)</div> <div style="border: 1px solid black; padding: 2px;">分钟 (00_H到59_H)</div> </div>	○	
Ⓕ + 5	被读取的日期 (一周)	日期 (一周) 以 BCD 码形式存储。 <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">b15 to b8</div> <div style="margin-right: 10px;">b7 to b0</div> </div> <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">00_H</div> <div style="border: 1px solid black; padding: 2px;">日期 (一周) (00_H到06_H)</div> </div> <div style="margin-left: 100px;"> 从 00_H (周日) 到 06_H (周六) </div>	○	

要点

当目标站的 QnACPU 的系统保护功能开启时（系统保护开关 SW5 状态为 ON），不能进行时钟数据的读/写操作。

(2) 在远程 RUN/STOP 期间

① 请求数据

软元件	条目	含义	远程 RUN	远程 STOP
Ⓔ	请求类型	0010 _H	○	○
Ⓔ + 1	子请求类型	0001 _H : 远程 RUN 0002 _H : 远程 STOP	○	○
Ⓔ + 2	模式	指定是否强制执行一次远程 RUN 操作。 0001 _H : 不强制运行 0003 _H : 强制运行（在远程 STOP 期间设定） 〔 如果一个处理了远程STOP操作的站不能执行一个远程RUN操作，那么将有另一个站来强制执行这个远程RUN操作。 〕	○	○
Ⓔ + 3	清除模式	在远程 RUN 期间指定 QnACPU 软元件内存状态。 0000 _H : 不清除（在远程 STOP 期间设定） 0001 _H : 清除（除锁存范围外） 0002 _H : 清除（包括锁存范围）	○	○

② 响应数据

软元件	条目	含义	远程 RUN	远程 STOP
Ⓕ	请求类型	0090 _H	○	○
Ⓕ + 1	子请求类型	0001 _H : 远程 RUN 0002 _H : 远程 STOP	○	○

[关于 CC-Link, A(1S)J71QC24]

CC-Link 可以用软件版本 J 或更新版本的主模块 (A(1S)J61QTB11)

(1) 在对时钟数据的读/写操作中

① 请求数据

软元件	条目	含义	读时钟数据	写时钟数据
⑤2	请求类型	0001 _H : 读时钟数据 0011 _H : 写时钟数据	○	○
⑤2 + 1	子请求类型	0002 _H : 读时钟数据 0001 _H : 写时钟数据	○	○
⑤2 + 2	更新格式 被更新的年份	指定将被写入的⑤2+3到⑤2+5的时钟数据的条目。 <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;"> b15 0 年(00_H到99_H) </div> <div style="margin-right: 10px;"> b7 0 </div> <div style="margin-right: 10px;"> b6 </div> <div style="margin-right: 10px;"> b5 </div> <div style="margin-right: 10px;"> b4 </div> <div style="margin-right: 10px;"> b3 </div> <div style="margin-right: 10px;"> b2 </div> <div style="margin-right: 10px;"> b1 </div> <div style="margin-right: 10px;"> b0 </div> </div> <div style="margin-top: 10px;"> 0: 不更新 1: 更新 </div> <div style="margin-top: 10px;"> → 年 → 月 → 日 → 小时 → 分钟 → 秒 → 日期(一周) </div>		○
⑤2 + 3	被更新的月份和日	月和日以 BCD 码形式存储 b15 到 b8 b7 到 b0 <div style="display: flex; justify-content: space-between;"> <div>日(01_H到31_H)</div> <div>月(01_H到12_H)</div> </div>		○
⑤2 + 4	被更新的小时和分钟	小时和分钟以 BCD 码形式存储 b15 到 b8 b7 到 b0 <div style="display: flex; justify-content: space-between;"> <div>分钟(00_H到59_H)</div> <div>小时(00_H到23_H)</div> </div>		○
⑤2 + 5	被更新的秒和日期 (一周)	秒和日期(一周)以 BCD 码形式存储 b15 到 b8 b7 到 b0 <div style="display: flex; justify-content: space-between;"> <div>日期(一周)(00_H到06_H)</div> <div>秒(00_H到59_H)</div> </div> <div style="margin-top: 10px;"> → 从00_H(星期天)到06_H(星期六) </div>		○

② 响应数据

软元件	条目	含义	读时钟数据	写时钟数据
⑤3	被读取的月份和年份	月份和年份(年份的最后两位)以 BCD 码形式存储。 b15 到 b8 b7 到 b0 <div style="display: flex; justify-content: space-between;"> <div>月(01_H到12_H)</div> <div>年(00_H到99_H)</div> </div>	○	
⑤3 + 1	被读取的小时和日期	小时和天以 BCD 码形式存储。 b15 到 b8 b7 到 b0 <div style="display: flex; justify-content: space-between;"> <div>小时(00_H到23_H)</div> <div>天(01_H到31_H)</div> </div>	○	
⑤3 + 2	被读取的秒和分钟	秒和分钟以 BCD 码形式存储。 b15 到 b8 b7 到 b0 <div style="display: flex; justify-content: space-between;"> <div>秒(00_H到59_H)</div> <div>分钟(00_H到59_H)</div> </div>	○	
⑤3 + 3	被读取的日期(一周)	日期(一周)以 BCD 码形式存储。 b15 到 b8 b7 到 b0 <div style="display: flex; justify-content: space-between;"> <div>00_H</div> <div>日期(一周)(00_H到06_H)</div> </div> <div style="margin-top: 10px;"> → 从00_H(周日)到06_H(周六) </div>	○	

(2) 远程 RUN/STOP

① 请求数据

软元件	条目	含义	远程 RUN	远程 STOP
⑤2	请求类型	0010 _H	○	○
⑤2 + 1	子请求类型	0001 _H : 远程 RUN 0002 _H : 远程 STOP	○	○
⑤2 + 2	模式	指定是否强制执行一次远程 RUN 操作。 0001 _H : 不强制运行 0003 _H : 强制运行 (在远程 STOP 期间设定) [如果一个处理了远程STOP操作的站不能执行一个远程RUN操作, 那么将有另一个站来强制执行这个远程RUN操作。]	○	○
⑤2 + 3	清除模式	在远程 RUN 期间指定 QnACPU 软元件内存状态。 0000 _H : 不清除 (在远程 STOP 期间设定) 0001 _H : 清除 (除锁存范围外) 0002 _H : 清除 (包括锁存范围)	○	○

② 响应数据

软元件	条目	含义	远程 RUN	远程 STOP
⑤3	请求类型	0090 _H	○	○
⑤3 + 1	子请求类型	0001 _H : 远程 RUN 0002 _H : 远程 STOP	○	○

要点

- (1) 当目标站的 QnACPU RUN/STOP 开关设为 RUN 时, 远程 RUN/STOP 功能被启动。
- (2) 当目标站的 QnACPU 的系统保护功能开启时 (系统保护开关 SW5 状态为 ON), 不能进行 RUN/STOP 操作。
- (3) 当目标站已经被另一个站设定在远程 STOP/PAUSE 模式下, 如果⑤2+1 模式没有设成 “不强制运行 (0001_H)”, 那么 RUN 操作不能被强制执行。
- (4) 如果处理远程 RUN/STOP 操作的目标站的 QnACPU 被复位, 那么远程 RUN/STOP 信息将丢失。

[功能]

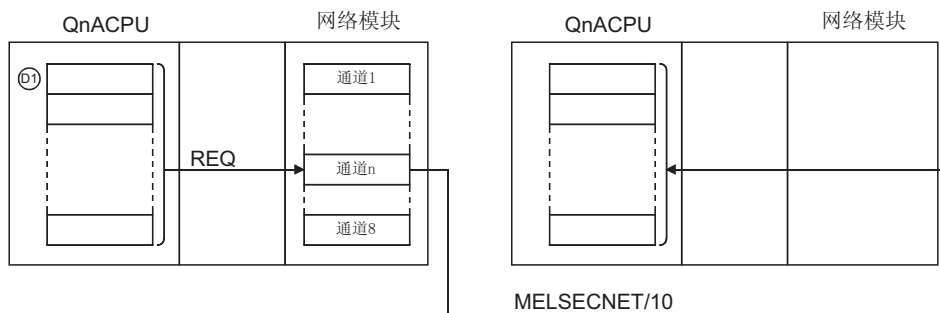
[对于 MELSECNET/10, A(1S)J71QE71]

- (1) 传送存储在②所指定的一些软元件中的请求数据，并处理服务请求，这些软元件属于一个连接到由控制数据中的网络号和站号指定的 MELSECNET/10 网络的站。

当来自目标站的软元件数据请求结束时，由②指定的结束软元件转为 ON 状态。

[自站]

[目标站]



- (2) 软元件数据的发送操作不但可以处理关于连接到指定网络号的 MELSECNET/10 的站，而且对于连接到自站网络的站也能处理。

- (3) 用于数据链接的指令不能在同一通道的一个以上的位置上同时执行。

当确定了指令在两个或多个位置同时执行的条件时，将自动处理握手操作，因此随后的用来数据链接的处理将不能完成。

- (4) 实际上 REQ 指令的执行还在进行当中，并且指令的正常结束和异常结束都可以由使用中的通道通讯指示标志，结束软元件②，和自站结束软元件 (②+1) 的状态显示来确定。

(a) 通讯指示标志

: 在执行 REQ 指令时转为 ON 状态，并且当发送结束时在扫描周期内为 END 处理时转为 OFF 状态。

(b) 自站结束软元件

: 当 REQ 指令的触发通讯结束时，在扫描周期内为 END 处理时转为 ON 状态，并且在下一个 END 处理时转为 OFF 状态。

(c) 处于结束时的状态显示软元件

: 由 REQ 指令结束时的状态来决定转为 ON 和 OFF 状态。

- 正常结束 : 保持 OFF 状态不变。
- 异常结束 : 当 REQ 指令结束时，在扫描周期内为 END 处理时转为 ON 状态，并且在下一个 END 处理时转为 OFF 状态。

[关于 CC-Link, A(1S)J71QC24]

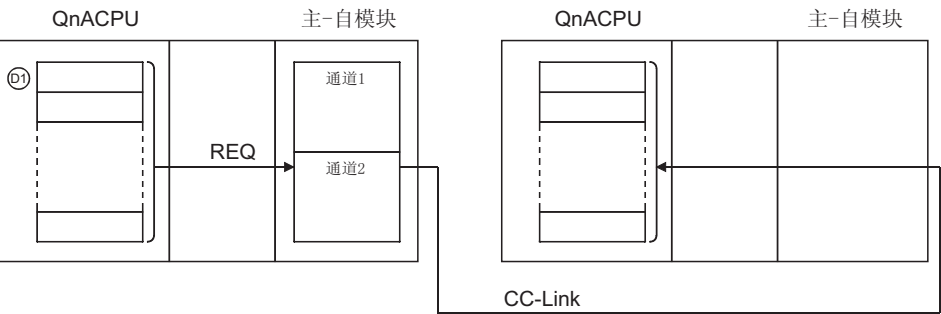
CC-Link 可以用软件版本 J 或更新版本的主模块 (A(1S)J61QTB11)

(1) 传送存储在②所指定的软元件开始的一些软元件中的请求数据，并处理服务请求。这些软元件属于一个连接到由控制数据中的站号指定的主站/自站号的站。

当来自目标站的软元件数据请求结束时，由②指定的结束软元件转为 ON 状态。

[自站]

[目标站]

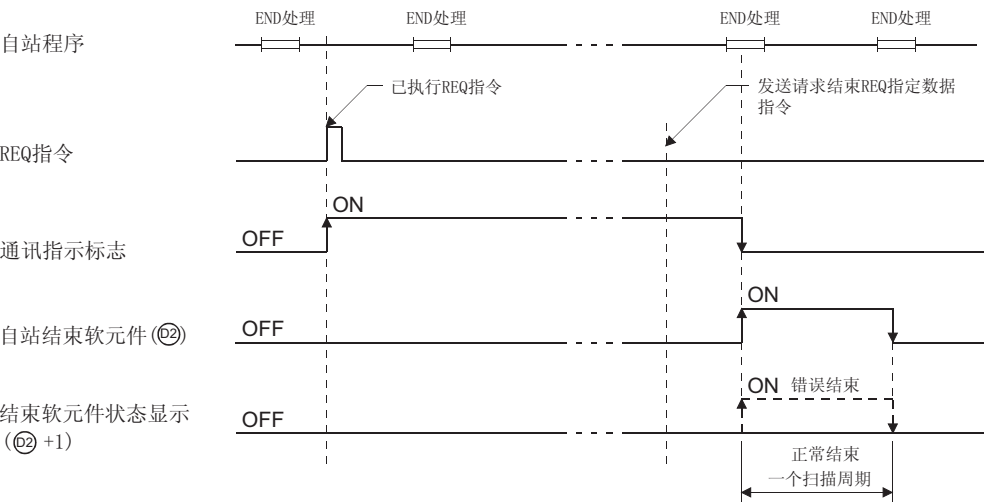


(2) 连接到 CC-Link 的站可以处理软元件数据的发送操作。

(3) 用于数据链接的指令不能在同一通道的一个以上的位置上同时执行。

当确定了指令在两个或多个位置同时执行的条件时，将自动处理握手操作，因此随后的用来数据链接的处理将不能完成。

[REQ 指令执行期间的自站操作]



[运行错误]

- (1) 当以下发生运行错误情况中，错误标志（SM0）变为 ON，并且错误代码存储在 SD0 中。
- 控制数据内容不在设定范围内(错误代码：4100)。
 - Jn 指定的网络号没有连接到这个站(错误代码：4102)。
 - Un 指定的 I/O 地址号不是一个网络模块(错误代码：2111)。

备注

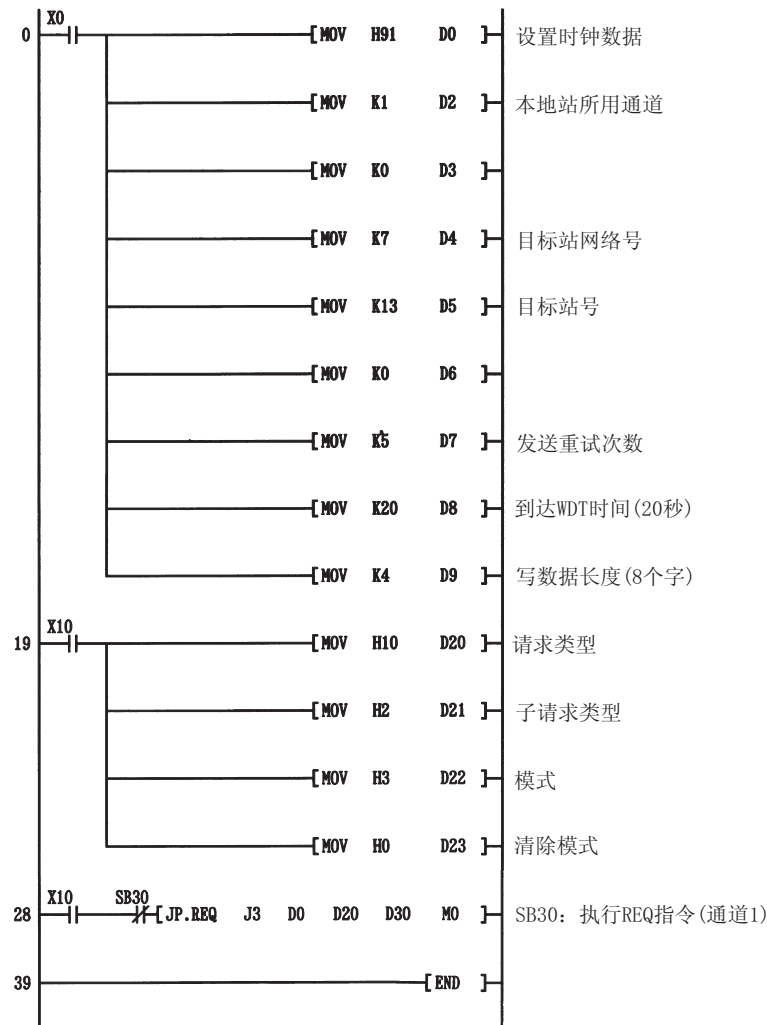
- 1) *1: 通道使用的通讯指示标志如下所示：
对于 CC-Link 和 A(1S)J71QC24，最多有两个通道可以使用。

通道号	1	2	3	4	5	6	7	8
通讯指示标志号	SB30	SB32	SB34	SB36	SB38	SB3A	SB3C	SB3E

[程序示例]

(1) 以下程序在 7 号网络的 13 号站的 QnACPU 上执行一条“STOP”操作。

[梯形图模式]



[列表模式]

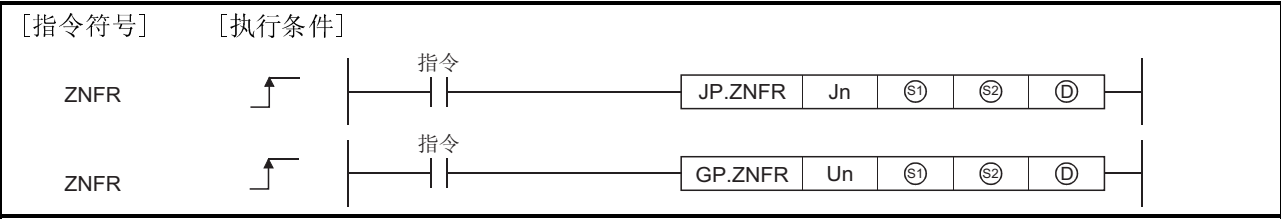
步	指令	软元件
0	LD	X0
1	MOV	H81
		D0
3	MOV	K1
		D2
5	MOV	K0
		D3
7	MOV	K7
		D4
9	MOV	K13
		D5
11	MOV	K0
		D6
13	MOV	K5
		D7
15	MOV	K20
		D8
17	LD	X10
18	MOV	H10
		D20
20	MOV	H2
		D21
22	MOV	H3
		D22
24	MOV	H0
		D23
26	LD	X10
27	ANI	S830
28	JP.REQ	J3
		D0
		D20
		D30
		M0
37	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×*3	×*3	×*3	○	○

8.2.8 从远程 I/O 站的特殊功能模块中读数据（ZNFR）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[]G[]		特殊功能模块 U[]G[]	变址寄存器 Zn	常数	其它
	位	字		位	字				
①	—	○	○	—					
②	—	○*	—	—					
④	○	—	—	—					

*：仅用于链接寄存器



[设定数据]

设定数据	含义	数据类型
Jn	自站的网络号 *1	BIN 16 位
Un	网络模块中自站的起始输入输出（I/O）号 *2	
①	存储控制数据的自站软元件的起始软元件号	软元件名称
②	在读操作完成后将要存储数据的本站链接寄存器(W)的起始软元件号	
④	在指令结束时，软元件变为 ON 一个扫描周期*3	位

备注

- *1: 自站的网络号指定如下：
- 1 到 239：网络号
 - 254：由设定为其它站可以访问的有效模块来指定的网络
- *2: 自站网络模块的起始输入输出（I/O）号可指定为从 0 到 FEH。
- *3: 每一个程序的局部软元件和文件寄存器不能被用作一个用于设定数据的软元件（具体细节请参照 QnACPU 编程手册（特殊功能模块））。

[控制数据]

软元件	条目	设定数据	设定范围	设定者
S1 + 0	执行类型	确认已到达 (位 0=1 (固定的))	0001 _H 0081 _H	用户
	错误的结束类型	当错误处理结束时设定时钟数据的设定状态。 不设定时钟数据 : 设定位 7 (b7) 为 0 设定时钟数据 : 设定位 7 (b7) 为 1 (S1+11 向前附加)		
S1 + 1	结束状态	存储指令的结束状态 0 : 没有出错 (正常结束) 其它不为 0 的 : 错误代码 *1	—	系统
S1 + 2	虚拟	未使用	—	—
S1 + 3	缓冲存储区地址	在缓冲存储区中指定起始地址	*2	用户
S1 + 4	虚拟	未使用	—	—
S1 + 5	目标站号	为读远程软元件数据的远程站设定站号	1 到 64	用户
S1 + 6	数字序列	在一系列特殊功能模块中指定安装在目标站的模块号	—	用户
S1 + 7	虚拟	未使用	—	—
S1 + 8	虚拟	未使用	—	—
S1 + 9	读数据长度	设定将要读取的数据数量	1 到 256	用户
S1 + 10	虚拟	未使用	—	—
S1 + 11	时钟数据标记 (仅在异常操作时设定)	存储时钟数据许可/禁止状态 当时钟数据为禁止状态时 : 0 当时钟数据为许可状态时 : 1	—	系统
S1 + 12	时钟数据 (仅在出错时设定)	高 8 位: 月 (1 到 12), 低 8 位: 年 (0 到 99)	—	系统
S1 + 13		高 8 位: 小时 (0 到 23), 低 8 位: 日 (1 到 31)		
S1 + 14		高 8 位: 秒 (0 到 59), 低 8 位: 分钟 (0 到 59)		
S1 + 15		高 8 位: 00 _H , 低 8 位: 星期几 (0 到 6)		

备注

*1: 有关错误代码请参照以下手册。

QnA/Q4AR MELSECNET/10 网络系统参考手册

*2: 请参照即将处理读操作的特殊功能模块使用手册。

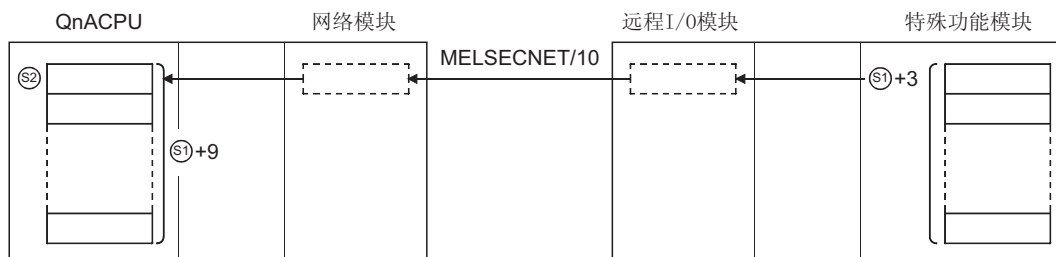
[功能]

- (1) 将由目标站控制数据所指定的 MELSECNET/10 远程 I/O 站的特殊功能模块缓冲存储区中的数据存储到由⑤₂指定的软元件开始的软元件中。

当完成了对来自远程 I/O 站的数据读操作时，由⑤指定的结束软元件变为 ON。

[自站（主站）]

[目标站（远程 I/O 站）]



- (2) 当一个远程站是来自 MELSECNET/10 网络的远程站时，读自这个远程 I/O 站的软元件数据只能被连接到同一网络的远程 I/O 站处理。

- (3) ZNFR 指令不能在同一特殊功能模块的两个位置上同时执行。

当确定了指令在两个或多个位置同时执行的条件时，将自动处理握手操作，因此已执行的 ZNFR 指令将不会被处理。

- (4) 在 ZNFR 指令的执行期内互锁信号广播读/写请求信号，读/写结束信号，一个结束软元件⑤，和结束显示软元件（⑤+1）的状态。

(a) 自站结束软元件

：在 ZNFR 指令结束时的扫描周期内，当为 END 处理信号时变为 ON，并且在下一个 END 处理时变为 OFF。

(b) 处于结束时的状态显示软元件

：由 ZNFR 指令结束时的状态来决定变为 ON 或 OFF。

● 正常结束 ：保持 OFF 状态不变。

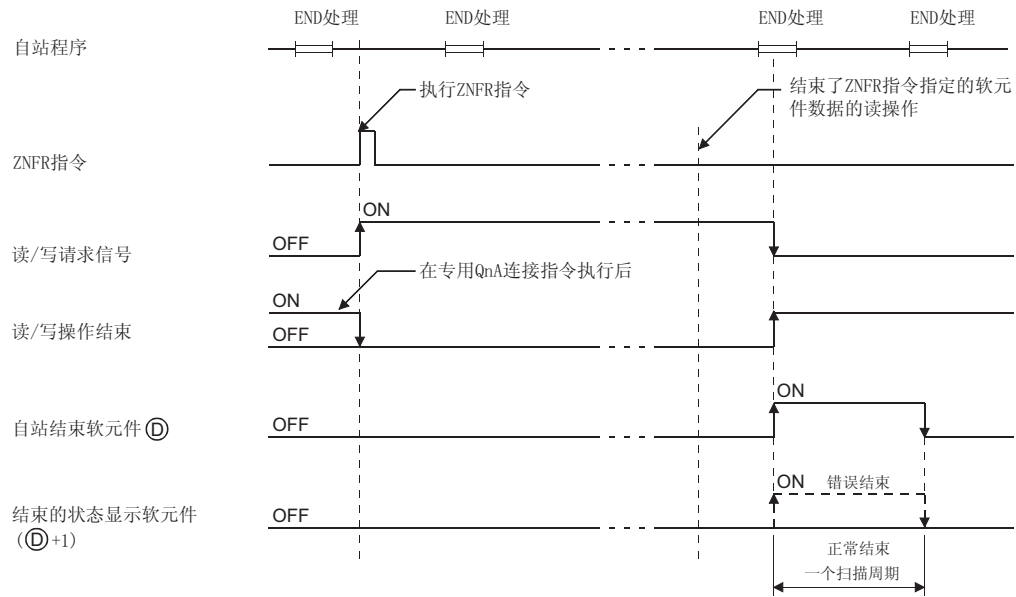
● 错误结束 ：在 ZNFR 指令结束的扫描周期内，当为 END 处理信号时变为 ON，并且在下一个 END 处理时变为 OFF。

(c) 每一个站的循环传送状态（SW74 到 77）

：网络中所有站的循环传送状态都以位信息形式存储，并且当一个站的循环传送发生异常网络中所有站的循环传送状态都以位信息形式存储，并且当一个站的循环传送发生异常，它的相应位被设定为 ON。

写一个程序以便 ZNFR 指令只在那些 ZNFR 执行目标站中位状态为 OFF（循环传送状态为正常）的站上执行。

[在 ZNFR 指令执行期间自站的操作]



(5) ⑤指定的链接寄存器 (W) 设为网络参数 “M ← R (从远程 I/O 站到主站)”，并且属于由链接刷新参数分配给 CPU 模块的范围内。

(6) 为了执行 ZNFR 指令，需要一个 OS 能使用的链接继电器和链接寄存器。

对于每个特殊功能模块，OS 要使用的链接继电器和链接寄存器号如下所示：

- 对 M → R (从主站到远程 I/O 站)..... 链接继电器：4，链接寄存器：4
- 对 M ← R (从远程 I/O 站到主站)..... 链接继电器：4，链接寄存器：4

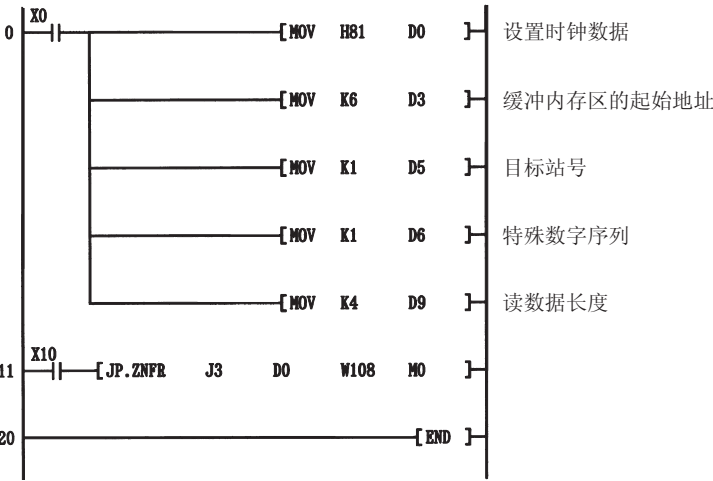
[运行错误]

(1) 在以下发生一个运行错误情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。

- 控制数据内容不在设定范围内 (错误代码：4100)。
- Jn 指定的网络号没有连接到这个站 (错误代码：4102)。
- Un 指定的 I/O 地址号模块不是一个网络模块 (错误代码：2111)。

[程序示例]

(1) 以下程序将网络号 3 到 W108 中 1R1 号站的起始特殊功能模块的缓冲内存地址 6 到 9 的内容读到 W10B 中去。
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	MOV	H81 D0
3	MOV	K6 D3
5	MOV	K1 D5
7	MOV	K1 D6
9	MOV	K4 D9
11	LD	X10
12	JP.ZNFR	J3 D0 W108 M0
20	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	×	×	○	○

8.2.9 往远程 I/O 站的特殊功能模块中写数据（ZNT0）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J□\□□		特殊功能模块 U□\G□	变址寄存器 Zn	常数	其它
	位	字		位	字				
①	—	○	○	—					
②	—	○*	—	—					
③	○	—	—	—					

*：仅用于链接寄存器

[指令符号]	[执行条件]
ZNT0	
ZNT0	
ZNT0	
ZNT0	

[设定数据]

设定数据	含义	数据类型
Jn	自站的网络号 *1	BIN 16 位
Un	网络模块中自站的起始输入输出(I/O)号 *2	
①	存储控制数据的自站软元件的起始软元件号	软元件名称
②	存储写入数据的自站链接寄存器(W)的起始软元件号	
③	在指令结束时，软元件变为 ON 一个扫描周期*3	位

备注

- *1: 自站的网络号指定如下：
 - 1 到 239：网络号
 - 254：由设定为其它站可以访问的有效模块来指定的网络
- *2: 自站网络模块的起始输入输出（I/O）号可指定为从 0 到 FEH。
- *3: 每一个程序的局部软元件和文件寄存器不能被用作一个用于设定数据的软元件（具体细节请参照 QnACPU 编程手册（特殊功能模块）。）

[控制数据]

软元件	条目	设定数据	设定范围	设定者
S1 + 0	执行类型	确认已到达 (位 0=1 (固定的))	0001 _H 0081 _H	用户
	错误的结束类型	当出错处理结束时设定时钟数据设定状态。 不设定时钟数据 : 设定位 7 (b7) 到 0 设定时钟数据 : 设定位 7 (b7) 到 1 (S1+11 向前附加)		
S1 + 1	结束状态	存储指令的结束状态 0 : 没有出错 (正常结束) 其它不为 0 的 : 错误代码 *1	—	系统
S1 + 2	虚拟	未使用	—	—
S1 + 3	缓冲存储区地址	在缓冲存储区中指定起始地址	*2	用户
S1 + 4	虚拟	未使用	—	—
S1 + 5	目标站号	指定目标站的站号	1 到 64	用户
S1 + 6	特殊功能模块数量	在一系列特殊功能模块中指定安装在目标站的模块号	—	用户
S1 + 7	虚拟	未使用	—	—
S1 + 8	虚拟	未使用	—	—
S1 + 9	写数据长度	设定将要写入的数据数量	1 到 256	用户
S1 + 10	虚拟	未使用	—	—
S1 + 11	时钟数据标记 (仅在异常操作时设定)	存储时钟数据的许可/禁止状态 当时钟数据为禁止状态时 : 0 当时钟数据为许可状态时 : 1	—	系统
S1 + 12	时钟数据 (仅当出错时设定)	高 8 位: 月 (1 到 12), 低 8 位: 年 (0 到 99)	—	系统
S1 + 13		高 8 位: 小时 (0 到 23), 低 8 位: 日 (1 到 31)		
S1 + 14		高 8 位: 秒 (0 到 59), 低 8 位: 分钟 (0 到 59)		
S1 + 15		高 8 位: 00 _H , 低 8 位: 星期几 (0 到 6)		

备注

*1: 有关错误代码请参照以下手册。

QnA/Q4AR MELSECNET/10 网络系统参考手册

*2: 请参照即将处理写操作的特殊功能模块的使用手册。

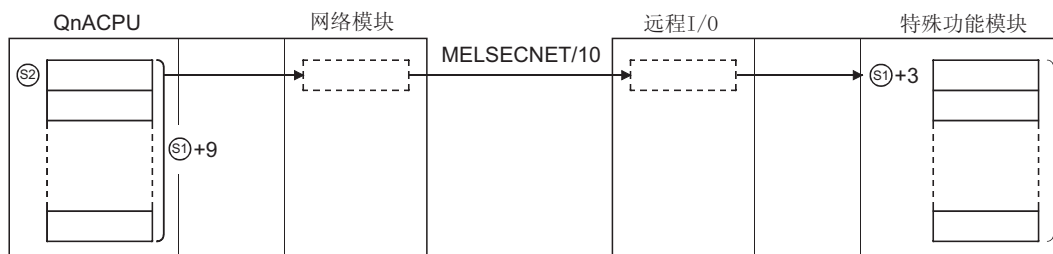
[功能]

- (1) 将由 $\textcircled{S2}$ 指定的软元件中存储的数据写到由目标站控制数据所指定的 MELSECNET/10 远程 I/O 站的特殊功能模块缓冲存储区中去。

当完成了将数据写到远程 I/O 站的操作时，由 $\textcircled{S1}$ 指定的结束软元件变为 ON。

[自站（主站）]

[目标站（远程 I/O 站）]



- (2) 只有当从一个远程 MELSECNET/10 主站写入一个连接到同一网络的远程 I/O 站时，对这个远程 I/O 站的写操作才能被执行。

- (3) ZNT0 指令不能在同一特殊功能模块的多个位置上同时执行。

当在多个位置同时执行指令的条件建立时，由于将自动执行了握手操作，因此 ZNT0 指令将不会被处理。

- (4) 在 ZNT0 指令的执行期间内的互锁信号包括一个读/写请求信号，一个读/写结束信号，结束软元件 $\textcircled{S1}$ ，和结束显示软元件 ($\textcircled{S1}+1$) 的状态。

- (a) 读/写请求信号

: 在一次读/写操作结束的扫描周期内，当 QnA 专用链接指令执行时状态变为 ON，END 信号处理时变为 OFF。

- (b) 读/写结束信号

: 在一次读/写操作结束的扫描周期内，当 QnA 专用链接指令执行时状态变为 OFF，END 信号处理时变为 ON。

- (c) 自站结束软元件

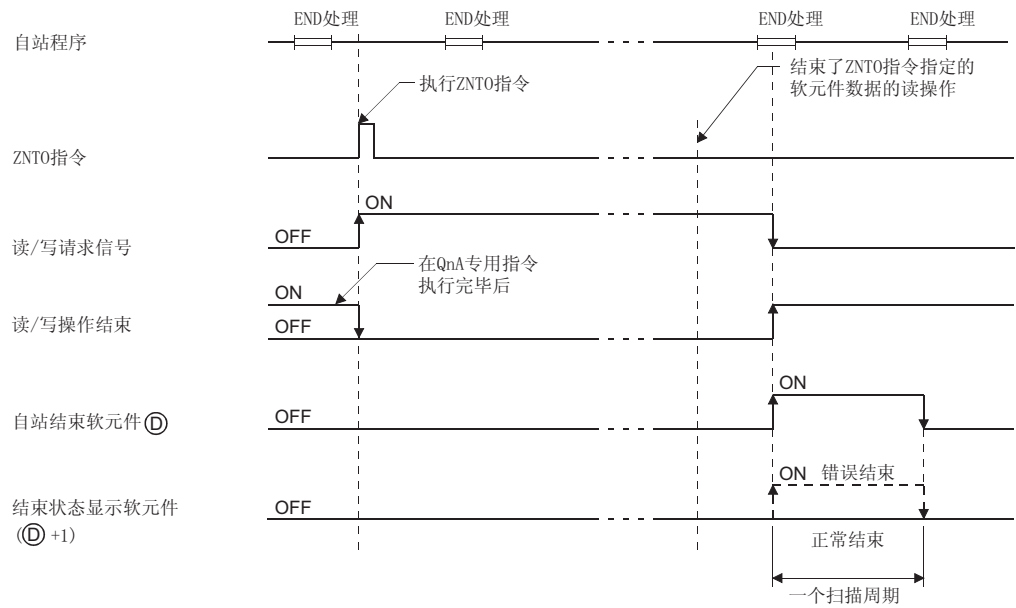
: 当 ZNT0 指令结束时，在扫描周期的 END 结束处理时变为 ON，并且在下一个 END 处理时变为 OFF。

- (d) 结束时的状态显示软元件

: 由 ZNT0 指令结束时的状态来决定变为 ON 或 OFF。

- 正常结束 : 保持 OFF 状态不变。
- 错误结束 : 当 ZNT0 指令结束时，在扫描周期的 END 结束处理时变为 ON，并且在下一个 END 处理时变为 OFF。

[在 ZNT0 指令执行期间自站的操作]



(5) ⑤②指定的链接寄存器 (W) 设定链接参数为“主机到远程”（从主站到远程 I/O 站），并且属于由链接刷新参数分配给 CPU 模块的范围内。

(6) 为了执行 ZNT0 指令，需要一个 OS 能使用的链接继电器和链接寄存器。

对于每个特殊功能模块，OS 要使用的链接继电器和链接寄存器号如下所示：

- 对 $M \rightarrow R$ （从主站到远程 I/O 站）..... 链接继电器：4, 链接寄存器：4
- 对 $M \leftarrow R$ （从远程 I/O 站到主站）..... 链接继电器：4, 链接寄存器：4

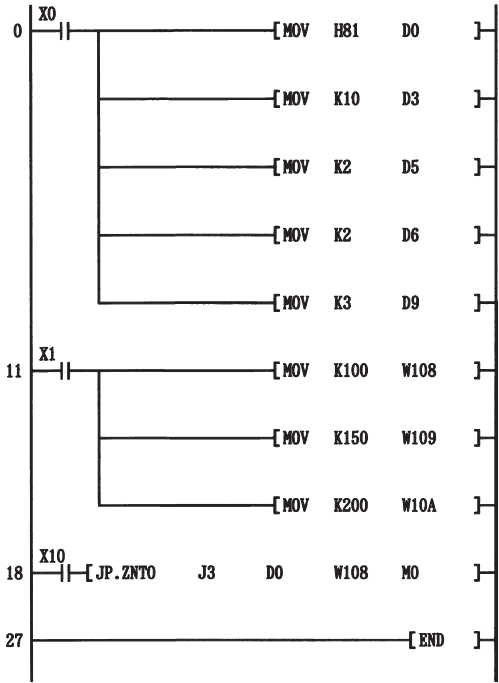
[运行错误]

(1) 当以下发生一个运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。

- 控制数据内容不在设定范围内 (错误代码：4100)。
- Jn 指定的网络号没有连接到这个站 (错误代码：4102)。
- Un 指定的 I/O 地址号模块不是一个网络模块 (错误代码：2111)。

[程序示例]

(1) 以下程序将 W108 到 W10A 中的内容写到 3 号网络 1R2 号站的第二特殊功能模块的 10 到 12 缓冲
内存地址中去。
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	MOV	H81
		D0
3	MOV	K10
		D3
5	MOV	K2
		D5
7	MOV	K2
		D6
9	MOV	K3
		D9
11	LD	X1
12	MOV	K100
		W108
14	MOV	K150
		W109
16	MOV	K200
		W10A
18	LD	X10
19	JP.ZNTO	J3
		D0
		W108
		M0
27	END	

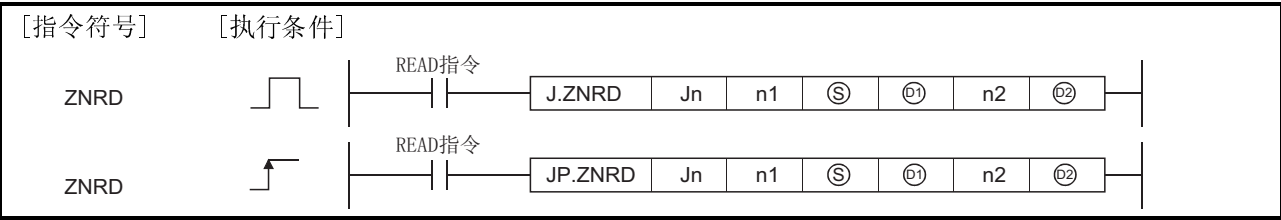
QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×*3	×*3	×*3	○	○

8.3 A 系列兼容链接指令

8.3.1 从其它站读软元件数据（MELSECNET/10）（ZNRD）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模块 U[][G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
n1	○	○	○	—			○	—	
⑤	—	△*1	—	—			—	—	
⑪	—	○	△*2	—			—	—	
n2	○	○	○	—			○	—	
⑪	○	○	○	—			—	—	

*1: 仅能使用 T, C, D 和 W
*2: 仅能用于 QnA



[设定数据]

设定数据	含义	数据类型
Jn	自站的网络号	BIN 16 位
n1	目标站号	
Ⓢ	存储读数据的站的起始软元件号	软元件名称
Ⓔ	存储读数据的自站的起始软元件号	
n2	接收数据长度	
Ⓔ	在指令结束时, 软元件变为 ON 的一个扫描周期	位

要点
(1) 当使用 ZNRD 指令从其它站读数据时, 自站和目标站两者之间软元件的参数设定必须相同。 (在其它站 CPU 模块Ⓢ中读取的起始软元件号)+(读取点数-1) ≤ (自站 CPU 模块的最终软元件号 *) *: 自站 CPU 模块的最后软元件号与Ⓢ有相同的软元件名。 (2) 每一个程序的局部软元件和文件寄存器不能被用作一个用于设定数据的软元件。 (具体细节请参照 QnACPU 编程手册 (特殊功能模块) 。)

备注

- *3 : 有关可用于 QCPU 的 READ 指令的相关细节请参照以下手册。
- Q 系列 MELSECNET/H 网络系统参考手册 (PLC 对 PLC 网络)
 - Q 系列以太网接口模块用户手册 (应用篇)

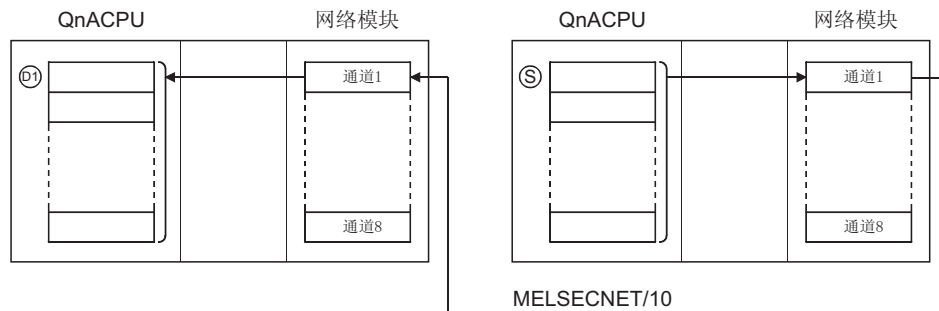
[功能]

- (1) 在 Jn 和 n1 指定的 MELSECNET/10 数据链接系统网络中，将由网络号⑤指定的软元件开始的一些软元件中的数据读到由⑥指定的软元件开始的一些软元件中去。

当来自目标站的软元件数据的读操作结束时，由⑥指定的结束软元件变为 ON 一个扫描周期。

[自站]

[目标站]



- (2) 当一个自站连接到 MELSECNET/10 系统中的一个网络时，只有连接到同一网络的站才能处理来自目标站的软元件数据的读操作。

注意必须设定路由参数。

- (3) Jn 指定从 1 到 239 的网络号。

0 指定 (J0) 等同于 MELSECNET 指定 (参照 8.3.2 部分)

- (4) 从 1 到 64 的站号可以在 n1 处指定。

- (5) 数据长度 (点数) 可由 n2 指定在 1 到 230 之间。

与 AnUCPU 不同，当 ACPU 被指定为目标站时，可被接受的请求数据的长度最多可达 32，但如果设定为 33 或更大的数据时也不会发生错误。

当在 n2 处指定为 0 时，不执行任何处理。

- (6) AnUCPU 站可以像 QnACPU 站一样处理由 ZNRD 指令触发的来自其它站的读操作。

- (7) 对同一通道而言，数据链接指令不能从多个位置同时执行。

当在两个或两个以上位置同时执行指令的条件建立时，由于自动处理握手操作，因此后续的用于数据链接指令处理将不能执行。

- (8) 对于 ZNRD 指令，自站和目标站都使用网络模块的通道 1。

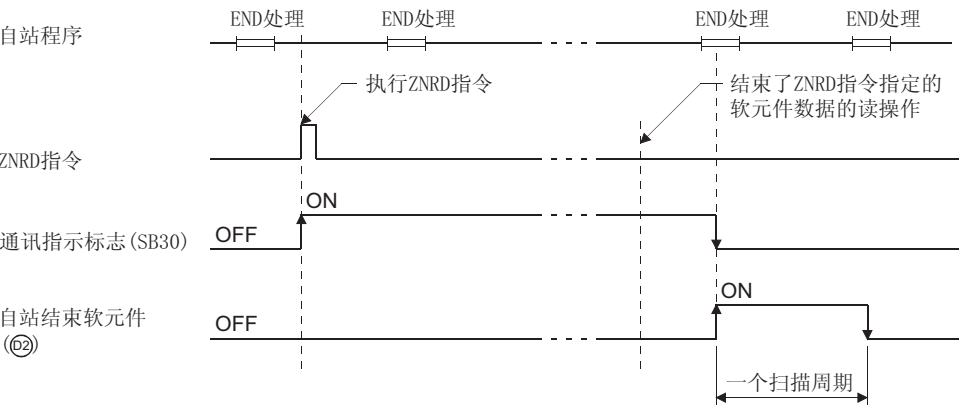
对一指定网络模块而言，使用多 ZNRD 指令的 QnA 专用指令和通道 1 不能同时执行。建立一个基于读/写请求信号和结束软元件的互锁，以便用于数据链接的多指令不会同时执行。

(9) 确认 ZNRD 指令正在执行，且一旦它结束，可使用 ZNRD 指示标志（SB30 和）结束软元件②来标记。

(a) ZNRD 指示标志
：在 ZNRD 指令执行时变为 ON，并且在读操作结束的扫描周期的 END 处理处变为 OFF。

(b) 自站结束软元件
：在 ZNRD 指令结束时的扫描周期内，当为 END 处理时变为 ON，且在下一个 END 处理时变为 OFF。

[在 ZNRD 指令执行期间自站的操作]



(10) ZNRD 指令的正常/错误结束可以用 ZNRD 指令处理结束寄存器（SW31）来确定。

正常结束时, SW31 为 0。
错误结束时, SW31 存储了适合的错误代码。*

[运行错误]

- (1) 在以下发生运行错误情况中，错误标志（SM0）变为 ON，并且错误代码存储在 SD0 中。
- 来自⑤指定的软元件的 n2 点超过了相应软元件范围的最终软元件号。 （错误代码：4101）
 - Jn 指定的网络号不存在。 （错误代码：4102）
 - n1 指定的站号不存在。 （错误代码：4102）
 - n2 指定的接收数据长度不在 1 到 230 的范围内。 （错误代码：4100）

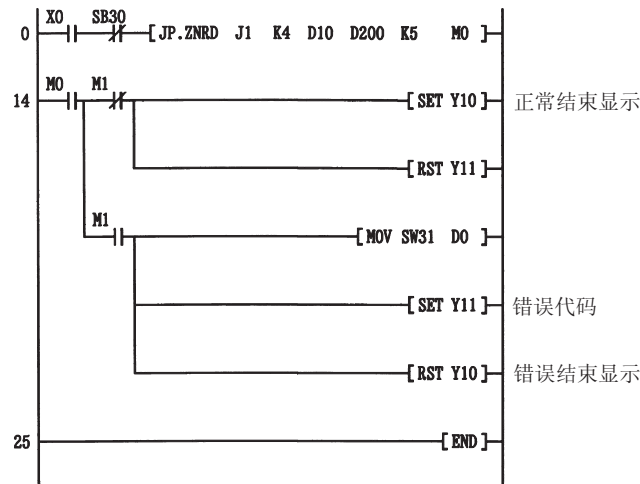
备注

*: 错误结束时的错误代码的具体细节请参照以下手册：
QnA/Q4AR MELSECNET/10 网络系统参考手册

[程序示例]

(1) 当 X0 为 ON 时，以下程序将 4 号站 D10 到 D14 的数据读到 1 号站的 D200 到 D204 中。

[梯形图模式]



[列表模式]

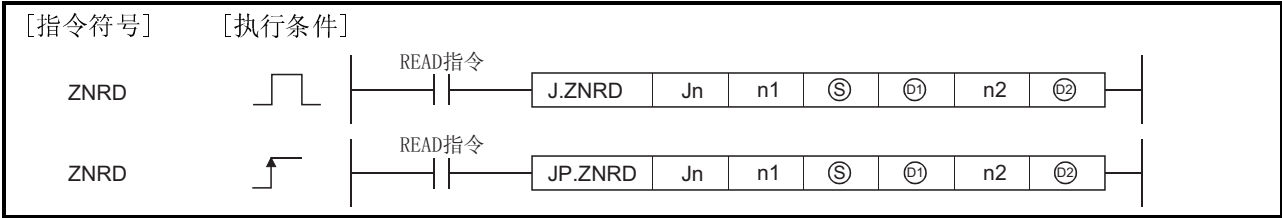
步	指令	软元件
0	LD	X0
1	ANI	SB30
2	JP.ZNRD	J1 K4 D10 D200 K5 M0
14	LD	M0
15	MPS	
16	ANI	M1
17	SET	Y10
18	RST	Y11
19	MPP	
20	AND	M1
21	MOV	SW31 D0
23	SET	Y11
24	RST	Y10
25	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	×	×	○	○

8.3.2 从本地站（MELSECNET）读软元件数据（ZNRD）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[]		特殊功能模块 U[]G[]	变址寄存器	常数 K, H	其它
	位	字		位	字				
n1	○	○	○	—			○	—	
⑤	—	△*	—	—			—	—	
①①	—	○	○	—			—	—	
n2	—	○	○	—			○	—	
①②	○	○	○	—			—	—	

*: 仅能使用 T, C, D 和 W



[设定数据]

设定数据	含义	数据类型
Jn	自站的网络号（固定为 0（J0））	BIN 16 位
n1	目标本地站号	
⑤	存储读数据的自站的起始软元件	软元件名称
①①	存储读数据的自站的起始软元件	
n2	接收数据长度	
①②	在指令结束时，软元件变为 ON 一个扫描周期	位

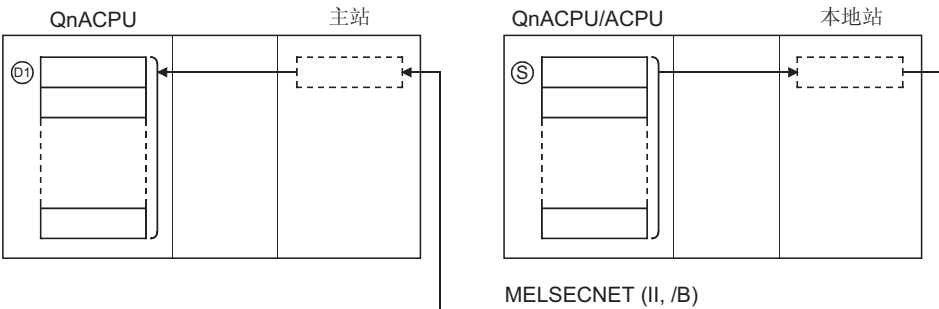
要点
(1) 当使用 ZNRD 指令从其它站读数据时，自站和目标站两者之间软元件的参数设定必须相同。 (在其它站 CPU 模块⑤中读取的起始软元件号) + (读取点数-1) ≤ (自站 CPU 模块的最终软元件号*) *: 自站 CPU 模块的最终软元件号与⑤有相同的软元件名。 (2) 每一个程序的局部软元件和文件寄存器不能被用作一个用来设定数据的软元件。 (具体细节请参照 QnACPU 编程手册（特殊功能模块）。)

[功能]

- (1) 在 MELSECNET 数据链接系统中，从 n1 指定的本地站中的由⑤指定的字软元件中读取 n2 点数据，并把它存储到本地站中②指定一些软元件中。
当来自自站的软元件数据的读操作结束时，在一个扫描周期中，由②指定的结束软元件变为 ON。

[自站]

[目标站]



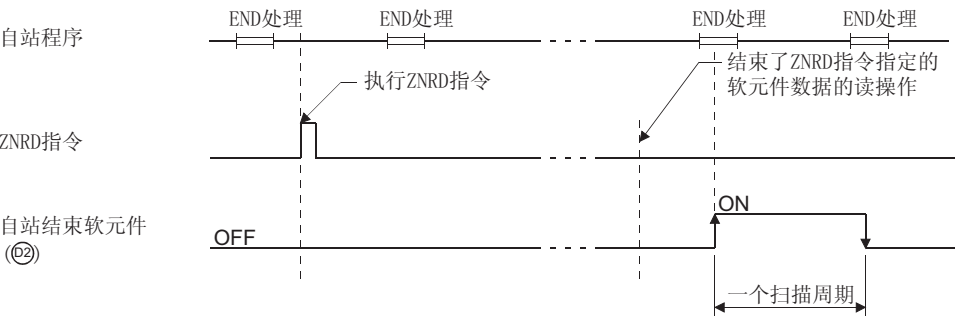
- (2) 来自本地站的软元件数据的读操作只能在 MELSECNET 主站中处理。
- (3) 在 MELSECNET 系统中，目标网络号 (Jn) 固定为 0 (J0)。
如果数字 1 到 239 用于 Jn，他们必须从 MELSECNET/10 中指定。(参照 8.3.1 部分)
- (4) 站号 1 到 64 在 n1 处指定。
当为 MELSECNET/B 时，1 到 31 的站号在 n1 处指定。

(5) ⑤指定的软元件范围如下所示

	MELSECNET 模式	MELSECNET II 模式
T	T0 到 T255	T0 到 T2047
C	C0 到 C255	C0 到 C1023
D	D0 到 D1023	D0 到 D6143
W	W0 到 W3FF	W0 到 WFFF

- (6) 接收数据长度 (点数) 1 到 32 在 n2 处指定。
当在 n2 处指定为 0 时，不执行任何进程。
- (7) ZNRD 和 ZNWR 指令不能同时执行。
当确定了指令在两个或两个以上位置同时执行的条件时，由于自动处理握手操作，因此后续 ZNRD 和 ZNWR 指令的处理将不能完成。
要执行多个 ZNRD 和 ZNWR 指令，采用结束软元件，并随后执行指令。
- (8) 当 ZNRD 结束时，在扫描周期的 END 指令处理处，结束软元件②变为 ON，并且在下一 END 处理处变为 OFF。

[在 ZNRD 指令执行期间自站的操作]



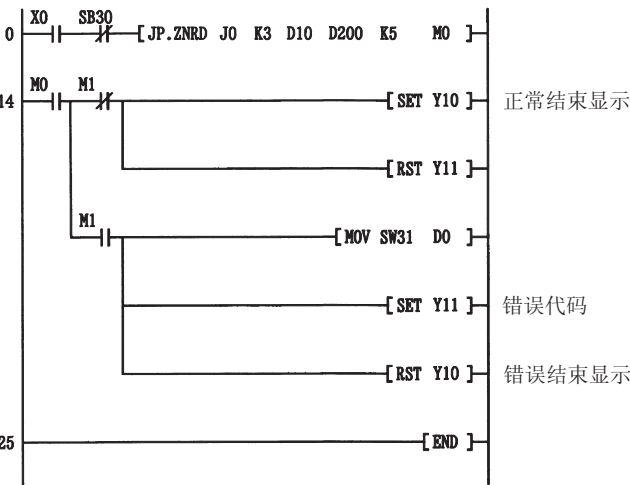
[运行错误]

- (1) 在以下发生运行错误情况中，错误标志（SM0）变为 ON，并且错误代码存储在 SD0 中。
- 来自Ⓔ指定的软元件的 n2 点超过了相应软元件范围的最终软元件号。 (错误代码：4101)
 - Jn 指定的网络号不存在。 (错误代码：4102)
 - n1 指定的站号不存在。 (错误代码：4102)
 - n2 指定的接收数据长度不在 1 到 32 的范围内。 (错误代码：4100)

[程序示例]

(1) 当主站的 X0 为 ON 时，以下程序将 3 号站 D10 到 D14 的数据读到主站的 D200 到 D204 中。

[梯形图模式]



[列表模式]

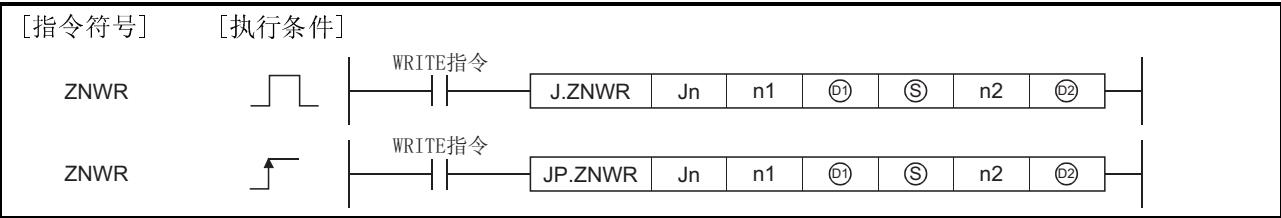
步	指令	软元件
0	LD	X0
1	ANI	SB30
2	JP.ZNRD	J0 K3 D10 D200 K5 M0
14	LD	M0
15	MPS	
16	ANI	M1
17	SET	Y10
18	RST	Y11
19	MPP	
20	AND	M1
21	MOV	SW31 D0
23	SET	Y11
24	RST	Y10
25	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×*3	×*3	×*3	○	○

8.3.3 将软元件数据写入其它站（MELSECNET/10）（ZNWR）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[]K[]		特殊功能模块 U[]V[]G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
n1	○	○	○	—			○	—	
①	—	△*1	—	—			—	—	
⑤	—	○	△*2	—			—	—	
n2	—	○	○	—			○	—	
②	○	○	○	—			—	—	

*1: 仅能使用 T, C, D 和 W
*2: 仅能用于 QnA



[设定数据]

设定数据	含义	数据类型
Jn	自站的网络号	BIN 16 位
n1	目标站号	
①	将要写入数据的目标站的起始软元件	软元件名称
⑤	正在存储写数据的自站的起始软元件	
n2	写入点数	
②	在指令结束时，软元件变为 ON 一个扫描周期	位

要点
(1) 当使用 ZNWR 指令从其它站写数据时，自站和目标站两者之间软元件的参数设定必须相同。 (在其它站 CPU 模块⑤中写入的起始软元件号)+(写入点数-1) ≤ (自站 CPU 模块的最终软元件号 *) *: 自站 CPU 模块的最后软元件号与⑤有相同的软元件名。 (2) 每一个程序的局部软元件和文件寄存器不能被用作一个用于设定数据的软元件。 (具体细节请参照 QnACPU 编程手册（特殊功能模块）。)

备注

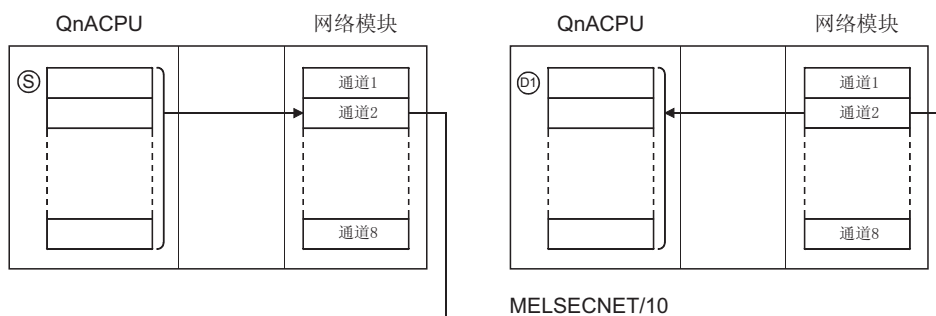
- *3 : 有关可用于 QCPU 的 ZNWR 指令的相关细节请参照以下手册。
- Q 系列 MELSECNET/H 网络系统参考手册（PLC 对 PLC 网络）
 - Q 系列以太网接口模块用户手册（应用）

[功能]

- (1) 在 MELSECNET/10 数据链接系统网络中，从自站的由⑤指定的字软元件中读取 n2 点数据，并将它储存到位于 Jn 和 n1 指定网络号的连接站中，起始软元件由⑥指定的一些软元件中去。
当对目标站的写操作结束时，由⑦指定的结束软元件变为 ON。

[自站]

[目标站]



- (2) 当一个自站连接到 MELSECNET/10 系统中的一个网络时，只有连接到同一网络的站才能处理对自站的软元件数据的写操作。
注意必须设定路由参数。
- (3) Jn 指定从 1 到 239 的网络号。
0 指定 (J0) 等同于 MELSECNET 指定 (参照 8.3.4 部分)
- (4) 从 1 到 64 的站号可以在 n1 处指定。
- (5) 传送数据长度可由 n 指定在 1 到 230 之间。
当在 n2 处指定为 0 时，不执行任何处理。
- (6) AnUCPU 目标站可以像 QnACPU 站一样处理由 ZNWR 指令触发的对其它站的写操作。
- (7) 对于自站和目标站，ZNWR 指令都使用网络模块的通道 1。对于一个指定网络模块，两条或两条以上的 ZNWR 指令，或者一条 ZNWR 指令和一条使用通道 2 的 QnA 链接专用指令都不能同时执行。
建立一个基于读/写请求信号和结束软元件的互锁，以避免两条或两条以上的数据链接的同时执行。

(8) 确认 ZNWR 指令正在执行，且一旦它结束，可使用 ZNWR 指示标志 (SB32 和) 结束软元件②来标记。

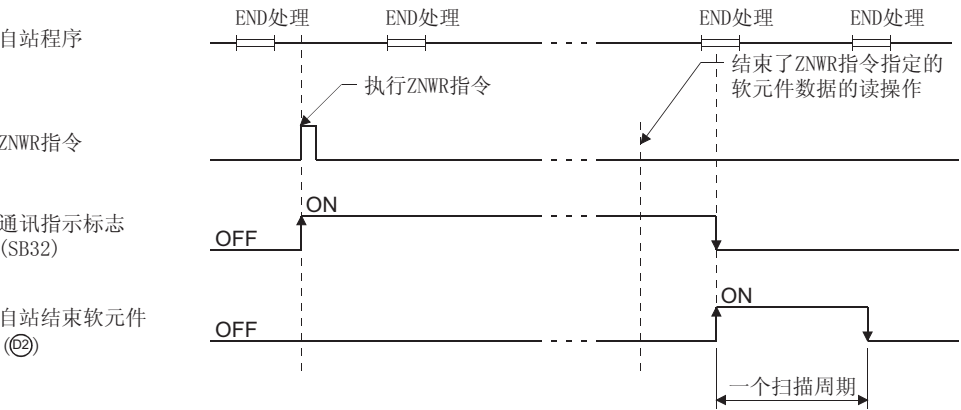
(a) ZNWR 指示标志

：在 ZNWR 指令执行时变为 ON，并且在写操作结束的扫描周期的 END 处理处变为 OFF。

(b) 本地站结束软元件

：在 ZNWR 指令结束时的扫描周期内，当为 END 处理时变为 ON，且在下一个 END 处理时变为 OFF。

[在 ZNWR 指令执行期间自站的操作]



(9) ZNWR 指令的正常/错误结束可以用 ZNWR 指令处理结束寄存器 (SW33) 来确定。

正常结束时, SW33 为 0。

错误结束时, SW33 存储了适合的错误代码。*

[运行错误]

- (1) 在以下发生运行错误情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- 来自⑤指定的软元件的 n2 点超过了相应软元件范围的最终软元件号 (错误代码：4101)。
 - Jn 指定的网络号不存在 (错误代码：4102)。
 - n1 指定的站号不存在 (错误代码：4102)。
 - n2 指定的接收数据长度不在 1 到 230 的范围内 (错误代码：4100)。

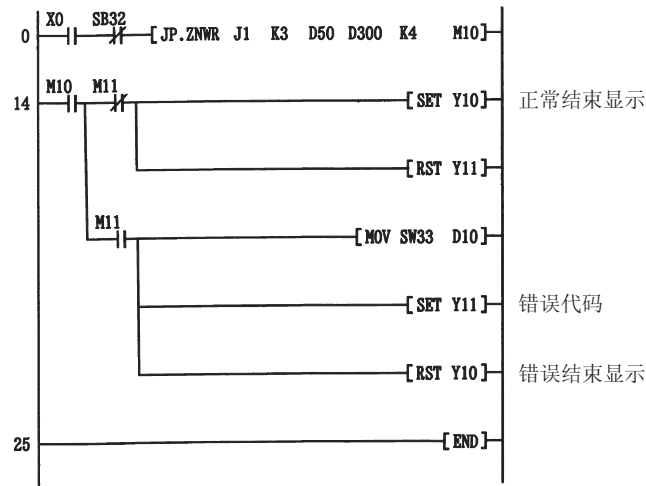
备注

- 1) *：错误结束时的错误代码的具体细节请参照以下手册：
QnA/Q4AR MELSECNET/10 网络系统参考手册

[程序示例]

(1) 当 2 号站的 X0 为 ON 时，以下程序将 2 号站 D300 到 D304 的数据写到 3 号站的 D50 到 D53 中。

[梯形图模式]



[列表模式]

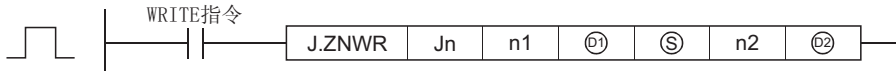
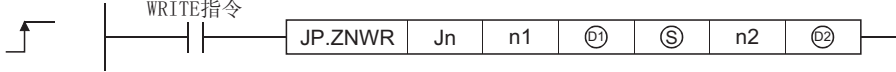
步	指令	软元件
0	LD	X0
1	ANI	SB32
2	JP.ZNWR	J1 K3 D50 D300 K4 M10
14	LD	M10
15	MPS	M10
16	ANI	M11
17	SET	Y10
18	RST	Y11
19	MPP	
20	AND	M11
21	MOV	SW33 D10
23	SET	Y11
24	RST	Y10
25	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	×	×	○	○

8.3.4 将数据写入本地站软元件中（MELSECNET）（ZNWR）

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[]\G[]		特殊功能模块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
n1	○	○	○	—		—	○
①	—	△*	—	—		—	—
⑤	—	○	○	—		—	—
n2	—	○	○	—		—	○
②	○	○	○	—		—	—

*: 仅能使用 T, C, D 和 W

[指令符号]	[执行条件]
ZNWR	
ZNWR	

[设定数据]

设定数据	含义	数据类型
Jn	目标站的网络号（固定为 0（J0））	BIN 16 位
n1	目标本站站号	
①	写入数据的本站站的起始软元件	软元件名
⑤	已储存写入数据的自站的起始软元件	
n2	接收数据长度	
②	在指令结束的一个扫描周期，软元件变为 ON 状态	位

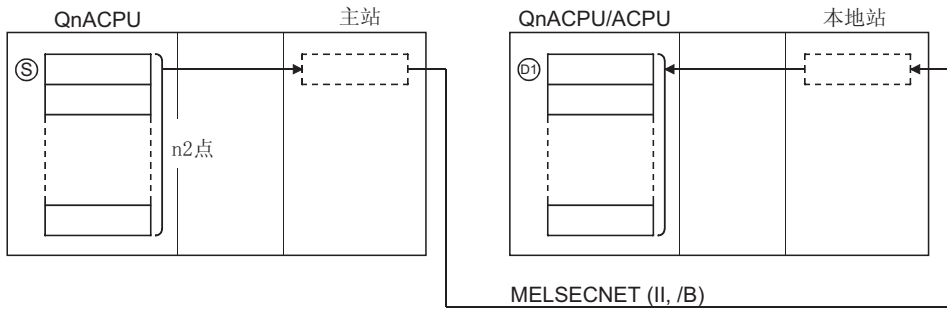
要点
(1) 当使用 ZNWR 指令从其它站读数据时，自站和目标站两者之间软元件的参数设定必须相同。 (在其它站 CPU 模块⑤中读取的起始软元件号)+(读取点数-1) ≤ (自站 CPU 模块的最终软元件号 *) *: 自站 CPU 模块的最终软元件号与⑤有相同的软元件名。
(2) 每一个程序的局部软元件和文件寄存器不能被用作一个用来设定数据的软元件。 (具体细节请参照 QnACPU 编程手册（特殊功能模块）。)

[功能]

- (1) 在 MELSECNET 数据链接系统中，从本地站的由⑤指定的字软元件中读取 n2 点数据，并将它储存到位于 n1 指定的本地站中、由⑥指定的软元件开始的一些软元件中去。
当对本地站的软元件数据的写操作结束时，在一个扫描周期中，由⑥指定的结束软元件变为 ON。

[自站]

[目标站]

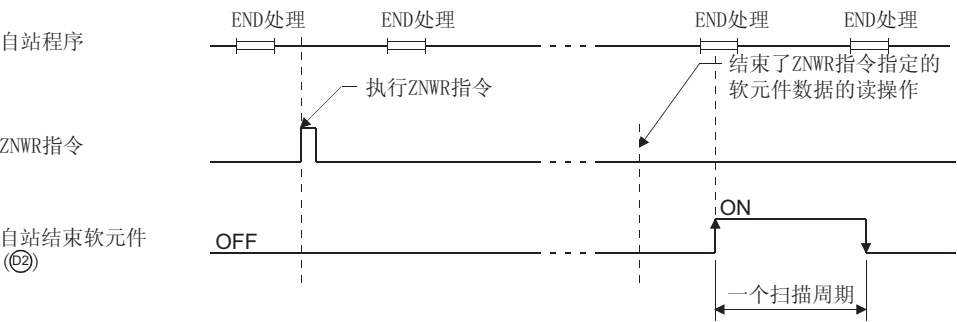


- (2) 对本地站的软元件数据的写操作只能在 MELSECNET 主站中处理。
- (3) 在 MELSECNET 系统中，目标网络号 (Jn) 固定为 0 (J0)。
如果 Jn 设为 1 到 239，这是作为一种 MELSECNET/10 指定。(参照 8.3.3 节)
- (4) 站号 1 到 64 可以在 n1 处指定。
在 MELSECNET/B 中，1 到 31 的站号在 n1 处指定。
- (5) ⑤指定的软元件范围如下所示。

	MELSECNET 模式	MELSECNET II 模式
T	T0 到 T255	T0 到 T2047
C	C0 到 C255	C0 到 C1023
D	D0 到 D1023	D0 到 D6143
W	W0 到 W3FF	W0 到 WFFF

- (6) n2 指定的传送数据长度 (点数) 为 1 到 32。
当在 n2 处指定为 0 时，不执行任何处理。
- (7) ZNRD 和 ZNWR 指令不能同时执行。
使用结束软元件预设一个互锁，以便 ZNRD 和 ZNWR 指令将不会同时执行。
当 ZNRD 和 ZNWR 指令执行一次以上时，是用一个结束软元件使得它们可以连续执行。
- (8) 当 ZNWR 指令结束时，在扫描周期的 END 指令处理处，结束软元件⑥变为 ON，并且在下一 END 处理处变为 OFF。

[在 ZNWR 指令执行期间自站的操作]

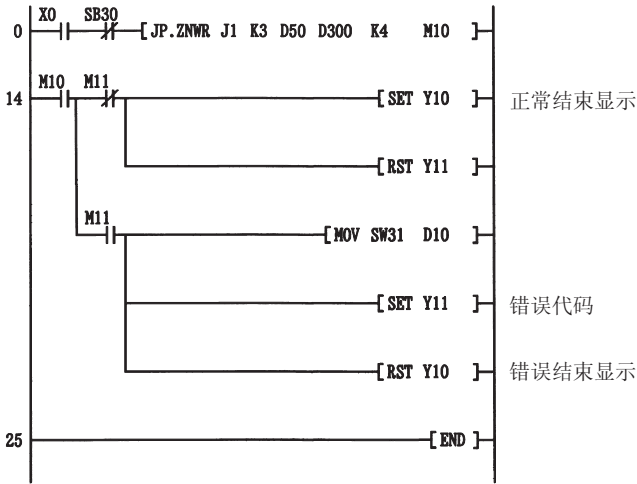


[运行错误]

- (1) 在以下发生运行错误情况中，错误标志（SM0）变为 ON，并且错误代码存储在 SD0 中。
- 来自⑤指定的软元件的 n2 点超过了相应软元件范围的最终软元件号 (错误代码：4101)。
 - Jn 指定的网络号不存在 (错误代码：4102)。
 - n1 指定的站号不存在 (错误代码：4102)。
 - n2 指定的接收数据长度不在 1 到 32 的范围内 (错误代码：4100)。

[程序示例]

- (1) 当主站的 X0 为 ON 时，以下程序将主站 D300 到 D303 的数据写到本地站的 D50 到 D53 中。
- [梯形图模式]



[列表模式]

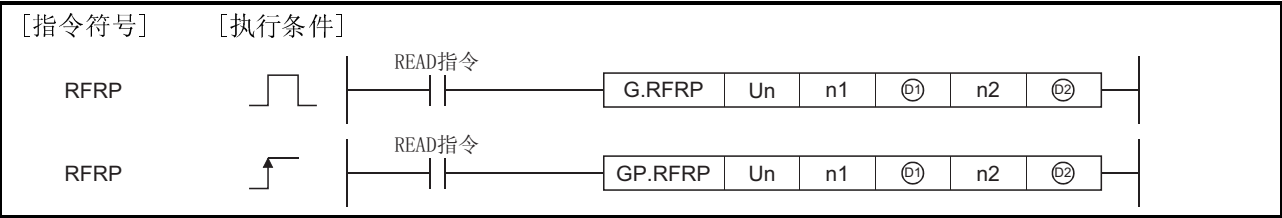
步	指令	软元件
0	LD	X0
1	ANI	SB30
2	JP.ZNWR	J1 K3 D50 D300 K4 M10
14	LD	M10
15	MPS	
16	ANI	M11
17	SET	Y10
18	RST	Y11
19	MPP	
20	AND	M11
21	MOV	SW31 D10
23	SET	Y11
24	RST	Y10
25	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	×	×	○	○

8.3.5 从一个远程 I/O 站特殊功能模块读数据（MELSECNET）（RFRP）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[][]		特殊功能模块 U[][G]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
n1	○	○	○	—			○	—	
①	—	△*	—	—			—	—	
n2	○	○	○	—			○	—	
②	○	○	○	—			—	—	

*: 仅能使用链接寄存器



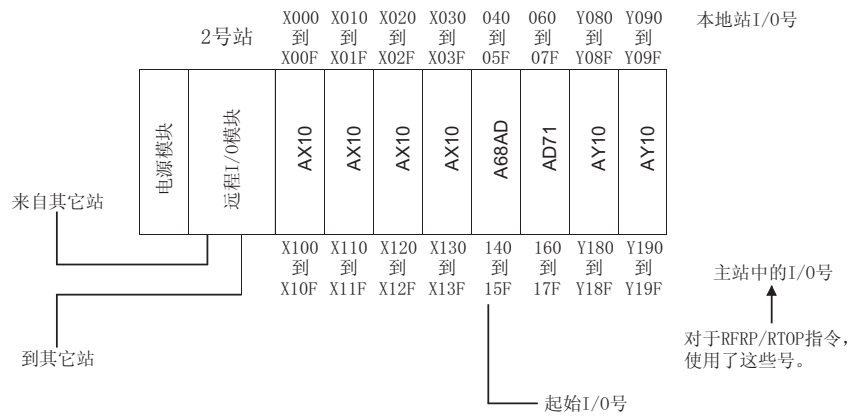
[设定数据]

设定数据	含义	数据类型
Un	为将执行数据写操作的特殊功能模块分配的主站的起始 I/O 号	BIN 16 位
n1	存储读数据的特殊功能模块的缓冲存储器的起始号	
①	将存储读数据的自站链接寄存器的起始软元件号	软元件名
n2	接收数据长度	BIN 16 位
②	在指令结束时, 软元件变为 ON 一个扫描周期	位

备注

- 来自主站的特殊功能模块的起始 I/O 号指定 Un。
- 每一个程序的局部软元件和文件寄存器不能被用作一个用于设定数据的软元件。（具体细节请参照 QnACPU 编程手册（特殊功能模块）。）

[示例]
假设 2 号远程 I/O 站的输入和输出分别分配为 X100 到 X17F 和 Y140 到 Y190。

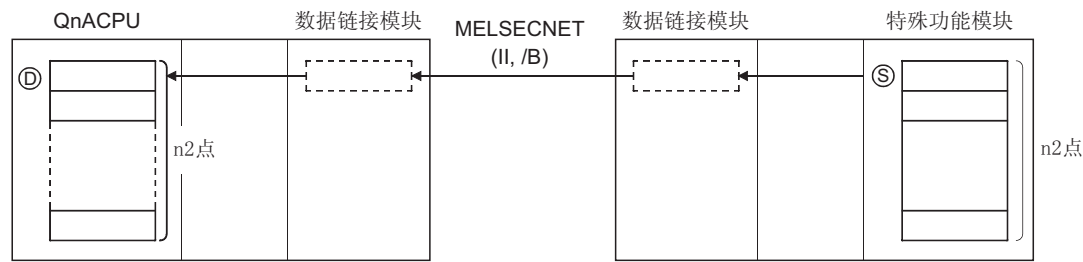


[功能]

- (1) 在 MELSECNET 数据链接系统中，将来自远程 I/O 站特殊功能模块分配给 Un 指定的 I/O 号的 n1 指定的缓冲存储器中的 n2 点数据，存储到本站（主站）的从③指定的软元件开始的一些软元件中去。
- 在来自远程 I/O 站的软元件数据的读操作结束的一个扫描周期内，②指定的结束软元件变为 ON。

[自站(主站)]

[目标站(远程I/O站)]

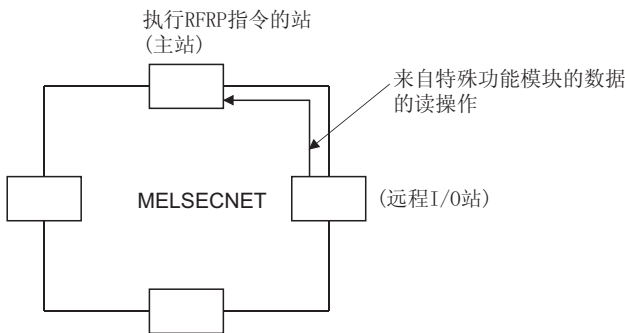


要点

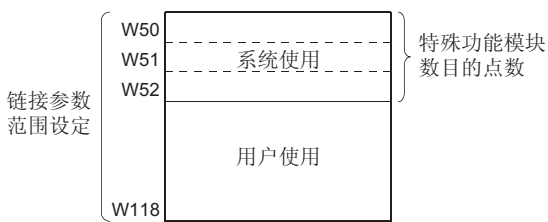
即使仅使用 RFRP 指令处理一个来自特殊功能模块的读操作，也必须设定系统在链接参数中将使用的“主站到远程 I/O 站”链接寄存器。

- (2) RFRP 指令执行标志 (YnE) 和结束标志 (X (n+1) E) 的自动 ON/OFF 控制，是由 RFRP 指令的内部处理进行的，因此用户不需要执行这些控制。

- (3) 来自远程 I/O 站软元件数据的读操作只能由 MELSECNET 主站处理。
而且，RFRP 指令的执行只能由占有 32 位 I/O 点的特殊功能模块进行。



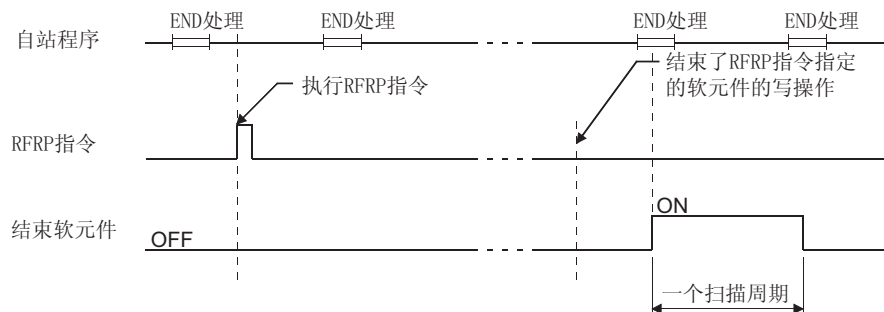
- (4) 当特殊功能模块发生一个错误且当 $X(n+1)D$ 为 ON 时执行 RFRP 指令， YnD 将自动变为 ON，并且执行一个错误复位操作。
但是，在错误复位处理中，不能处理通讯处理。
- (5) 当 I/O 号扩展为 4 位时，将 Un 指定的特殊功能模块起始 I/O 号设为高 3 位。
例如，对于 $X/Y0200$ ，设定为 020。
- (6) 关于 $n1$ 指定的特殊功能模块缓冲存储器的附加信息，请参照特殊功能模块使用手册。
- (7) $n2$ 处的发送数据长度（点数）可设为 1 到 16。
- (8) ②指定的链接寄存器设定范围选择为链接参数“从主站到远程 I/O 站”。
- (9) 在参数“从主站→远程 I/O 站”中设定的链接寄存器中，从第一个链接寄存器开始的区域，和安装在相关远程 I/O 站对应的特殊模块数，都为系统使用。
例如，连接参数中“从主站→远程 I/O 站”设定为 $W50$ 到 $W118$ ，则由于所讨论的安装在远程 I/O 站有 3 个特殊功能模块， $W50$ 到 $W52$ 将保留给系统使用。



- (10) 对同一特殊功能模块，RFRP 和 RTOP 指令不能同时执行。
当在两个或多个位置同时执行指令的条件建立时，将自动处理握手操作，因此随后执行的 RFRP 或 RTOP 指令将不会被处理。
如果对同一特殊功能模块既执行 RFRP 又执行 RTOP 指令，则使用一个结束软元件以便他们能连续的执行。

- (11) 在 RFRP 指令结束时的扫描周期内，当⑤为 END 处理信号时结束软元件变为 ON，并且在下一个 END 处理时变为 OFF。

[在 RFRP 指令执行期间自站的操作]



[运行错误]

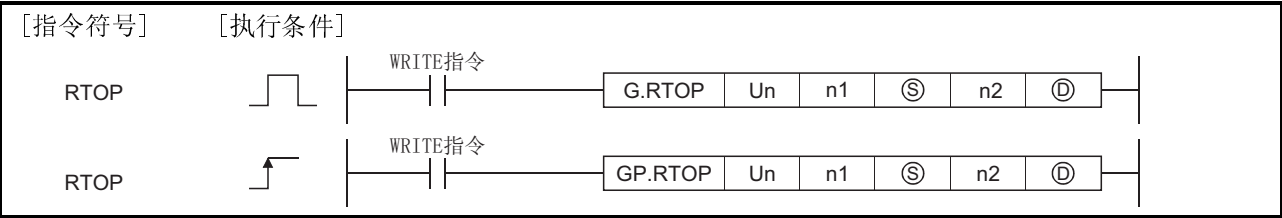
- (1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- n1 指定的 I/O 号与远程 I/O 站的 I/O 号不一致 (错误代码: 4102)。
 - n1 指定的 I/O 号不是一个特殊功能模块的起始 I/O 号 (错误代码: 4102)。
 - 来自⑤指定软元件的 2 点区超过了相关软元件范围的最终软元件号 (错误代码: 4101)。
 - Un 指定的模块不存在 (错误代码: 2413)。
 - n2 指定的点数超过 1 到 16 这个范围 (错误代码: 4100)。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	×	×	○	○

8.3.6 将数据写入远程 I/O 站特殊功能模块（MELSECNET）（RTOP）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[]		特殊功能模块 U[]G[]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
n1	○	○	○	—			○	—	
⑤	—	△*	—	—			—	—	
n2	○	○	○	—			○	—	
⑥	○	○	○	—			—	—	

※: 仅能使用链接寄存器



[设定数据]

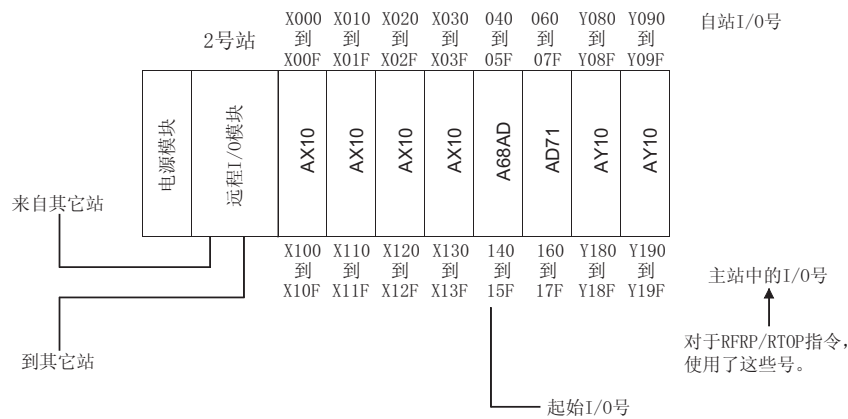
设定数据	含义	数据类型
Un	为将执行数据写操作的特殊功能模块分配的主站的起始 I/O 号	BIN 16 位
n1	存储写数据的特殊功能模块的缓冲存储器的起始号	
⑤	存储写数据的自站链接寄存器的第一个软元件号	软元件名称
n2	传送数据长度	BIN 16 位
⑥	在指令结束时, 软元件变为 ON 一个扫描周期	位

备注

- 由主站的特殊功能模块的起始 I/O 号指定 Un。
- 每一个程序的局部软元件和文件寄存器不能被用作一个用于设定数据的软元件。（具体细节请参照 QnACPU 编程手册（特殊功能模块））。

[示例]

假设 2 号远程 I/O 站的输入和输出分别分配为 X100 到 X17F 和 Y140 到 Y190。

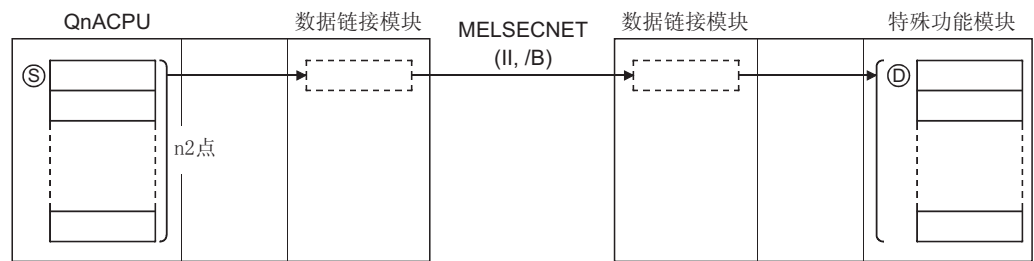


[功能]

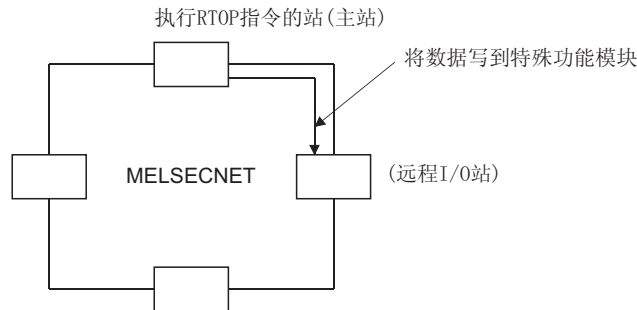
- (1) 在⑤指定的 MELSECNET 数据链接系统中，存储来自自站（主站）的 n2 点数据，随后的远程 I/O 站的特殊功能模块中⑥指定的缓冲存储器分配给 Un 指定的 I/O 号。
当对远程 I/O 站的软元件写操作结束后，在一个扫描周期内，⑥指定的结束软元件变为 ON。

[自站(主站)]

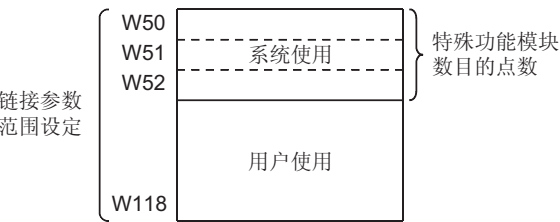
[目标站(远程I/O站)]



- (2) RTOP 指令执行标志 (YnF) 和结束标志 (X (n+1) F) 的自动 ON/OFF 控制，是由 RTOP 指令的内部处理进行的，因此用户不需要执行这些处理。
- (3) 来自远程 I/O 站软元件数据的写操作只能由 MELSECNET 主站处理。
更进一步，RTOP 指令的执行只能由占有 32 位 I/O 点的特殊功能模块进行。



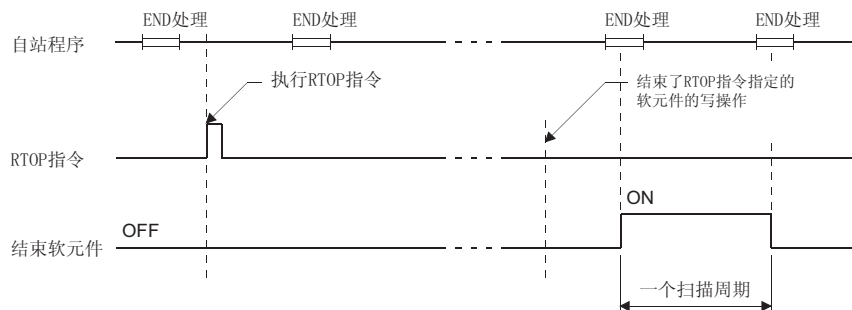
- (4) 当特殊功能模块发生一个错误且 X (n+1) D 变为 ON 时，执行 RTOP 指令，YnD 将自动变为 ON，并且执行一个错误复位操作。
但是，在错误复位处理中，不能进行通讯处理。
- (5) 当 I/O 号扩展为 4 位时，将 Un 指定的特殊功能模块起始 I/O 号设为高 3 位。
例如，对于 X/Y0200, 设定为 020。
- (6) 关于 n1 指定的特殊功能模块缓冲存储器的附加信息，请参照特殊功能模块使用手册。
- (7) n2 处的发送数据长度（点数）可设为 1 到 16。
- (8) ⑤指定的链接寄存器的范围在链接参数“从主站到远程 I/O 站”的选择中设定。
而在链接寄存器中，从由链接参数选择中设定的“从主站到远程 I/O 站”开始处到包含装在相关远程 I/O 站上的特殊模块个数为上的区域是系统保留使用的。
例如，连接参数中“从主站→远程 I/O 站”设定为 W50 到 W118，则由于所讨论的安装在远程 I/O 站有 3 个特殊功能模块，W50 到 W52 将保留给系统使用。



- (9) 对同一特殊功能模块，RFRP 和 RTOP 指令不能同时执行。
当在两个或多个位置同时执行指令的条件建立时，将自动处理握手操作，因此随后执行的 RFRP 和 RTOP 指令将不会被处理。
如果同一特殊功能模块希望执行多 RFRP/RTOP 指令，使用一个结束软元件以便他们将连续的执行。

(10) 在 RTOP 指令结束时的扫描周期内，当⑤为 END 处理信号时结束软元件变为 ON，并且在下一个 END 处理时变为 OFF。

[在 RTOP 指令执行期间自站的操作]



[运行错误]

(1) 在以下发生运行错误的情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。

- n1 指定的 I/O 号与远程 I/O 站的 I/O 号不一致 (错误代码: 4102)。
- n1 指定的 I/O 号不是一个特殊功能模块的起始 I/O 号 (错误代码: 4102)。
- 来自⑤指定软元件的 2 点区超过了相关软元件范围的最终软元件号 (错误代码: 4101)。
- Un 指定的模块不存在 (错误代码: 2413)。
- n2 指定的点数超过 1 到 16 这个范围 (错误代码: 4100)。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

8.4 读/写路由信息

8.4.1 读路由信息（RTREAD）

(1) 使用高性能型 QCPU 和过程控制 CPU 时

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
n	○	○		—			○
Ⓓ	—	○		—			—



[设定数据]

设定数据	含义	数据类型
n	传送指定网络号（1 到 239）	BIN 16 位
Ⓓ	将存储读数据的软元件的起始软元件号	软元件名称

[功能]

- (1) 从由 n 指定目标网络读取数据，使用在路由参数中设定的路由信息，并且存储到从Ⓓ开始的软元件中去。
- (2) 如果在路由参数中设定的 n 所指定的传输目标网络中没有数据，则将 0 存储到从Ⓓ开始的软元件中去。
- (3) 从Ⓓ开始的软元件中存储的数据内容如下所示。

(单个数据范围)

Ⓓ+0	中继网络号	(1到239)
+1	中继站号	(1到64)
+2	虚拟	

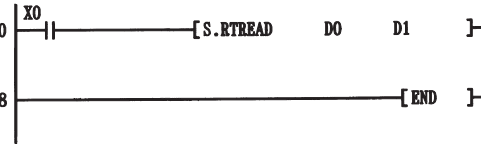
[运行错误]

- (1) 在以下发生运行错误情况中，错误标志（SM0）变为 ON，并且错误代码存储在 SD0 中。
 - n 指定的数据不在 1 到 239 的范围内。（错误代码：4100）

[程序示例]

(1) 当 X0 为 ON 时，以下程序从 D0 指定网络号中读取路由信息。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	S.RTREAD	D0 D1
8	END	

[操作]

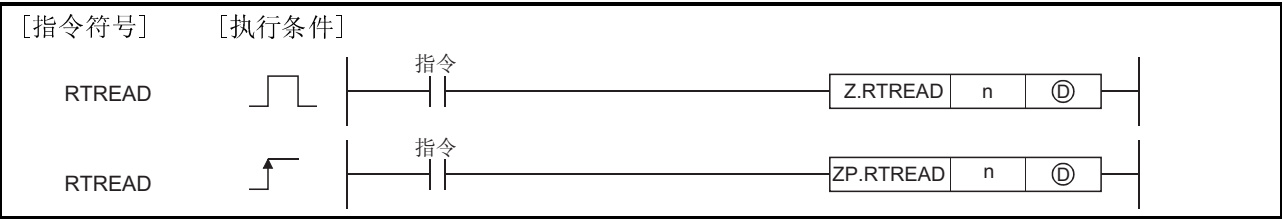
D0	1
D1	10
D2	3
D3	虚拟

[路由参数设定内容]

发送指定网络号	中继网络号	中继网络号
1	10	3
2	10	2
3	10	1

(2) 使用 QnACPU 时

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 JEJG		特殊功能模块 UJG	变址寄存器 Zn
	位	字		位	字		
n	○	○		—			○
Ⓓ	—	○		—			—



[设定数据]

设定数据	含义	数据类型
n	传送指定网络号 (1 到 239)	BIN 16 位
Ⓓ	将存储读数据的软元件的起始软元件号	软元件名称

[功能]

- (1) 从由 n 指定的目标网络中读取数据，使用在路由参数中设定的路由信息，并且存储到从Ⓓ开始的软元件中去。
- (2) 如果在路由参数中设定的 n 所指定的传输目标网络中没有数据，则将 0 存储到从Ⓓ开始的软元件中去。
- (3) 从Ⓓ开始的软元件中存储的数据内容如下所示。

(单个数据范围)

Ⓓ+0	继电器网络号	(1到239)
+1	继电器站号	(1到64)
+2	路由站号	(1到64)

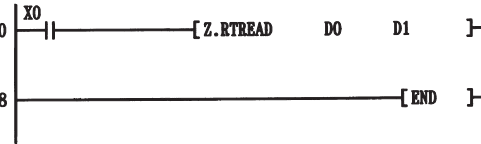
[运行错误]

- (1) 在以下发生运行错误情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- n 指定的数据不在 1 到 239 的范围内。(错误代码: 4100)

[程序示例]

(1) 当 X0 为 ON 时，以下程序从 D0 指定网络号中读取路由信息。

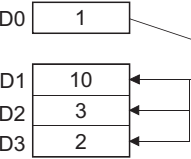
[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	Z.RTREAD	D0 D1
8	END	

[操作]



[路由参数设定内容]

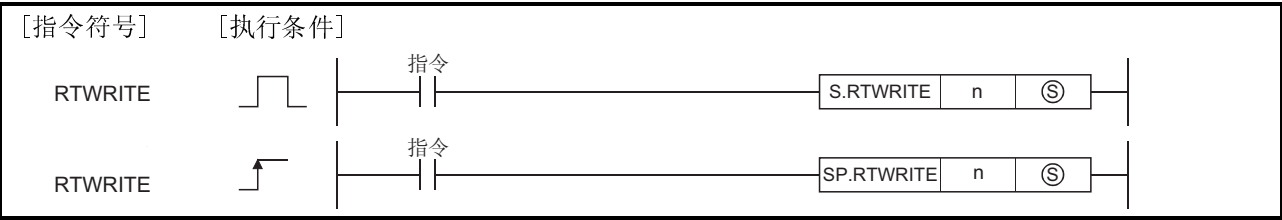
发送指定网络号	发送指定网络号	中继器站号	路由站号
1	10	3	2
2	10	2	2
3	10	1	2

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	○	○

8.4.2 登录路由信息（RTWRITE）

(1) 使用高性能型 QCPU 和过程控制 CPU 时

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
n	○	○		—			○
Ⓢ	—	○		—			—



[设定数据]

设定数据	含义	数据类型
n	传送指定网络号（1 到 239）	BIN 16 位
Ⓢ	存储写数据的软元件的起始软元件号	软元件名称

[功能]

(1) 路由参数中，由 n 所指定的目标网络的寄存器路由数据从Ⓢ开始。

(单个数据范围)

Ⓢ+0	中继网络号	(1到239)
+1	中继站号	(1到64)
+2	虚拟	

(2) 如果 n 指定的目标网络数据在路由参数中设定，它将用于从Ⓢ开始的数据的更新。

(3) 如果所有从Ⓢ开始的数据为 0，n 指定的指定网络数据将从路由参数中删除。

[运行错误]

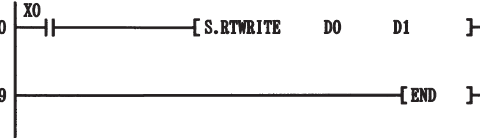
(1) 在以下发生运行错误的情况中，错误标志（SM0）变为 ON，并且错误代码存储在 SD0 中。

- n 指定的数据不在 1 到 239 的范围内(错误代码：4100)。
- 从Ⓢ开始的数据超过了单个设定范围(错误代码：4100)。
- 登记在网络参数的路由参数中的路由数据和登记在 RTWRITE 指令中的路由数据的总和超过了 64(错误代码：4100)。

[程序示例]

(1) 当 X0 为 ON 时，以下程序将 D1 到 D3 指定的路由信息写到 D0 指定网络号的网络模块中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	S.RTWRITE	D0 D1
9	END	

[操作]

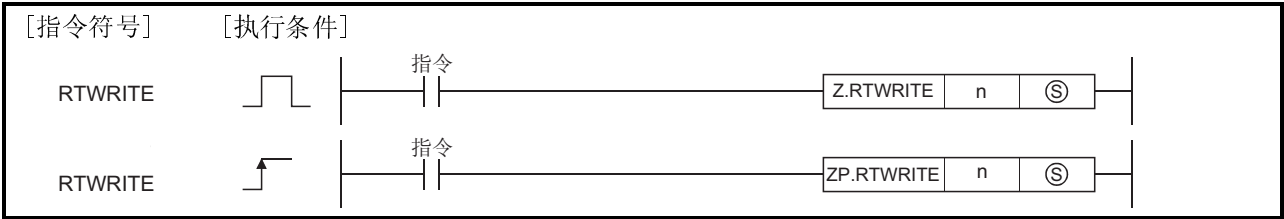
D0	1
D1	20
D2	1
D3	虚拟

[路由参数设定内容]

发送指定网络号	中继网络号	中继站号
1	20	1
2	10	2
3	10	1

(2) 使用 QnACPU 时

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
n	○	○		—			○
Ⓢ	—	○		—			—



[设定数据]

设定数据	含义	数据类型
n	传送指定网络号 (1 到 239)	BIN 16 位
Ⓢ	存储写数据的软元件的起始软元件号	软元件名称

[功能]

- (1) 在路由参数中，由 n 所指定的目标网络的寄存器路由数据从Ⓢ开始。
- (2) 如果 n 指定的目标网络数据在路由参数中设定，它将用于从Ⓢ开始的数据的更新。
- (3) 如果所有从Ⓢ开始的数据为 0，n 指定的指定网络数据将从路由参数中删除。

[运行错误]

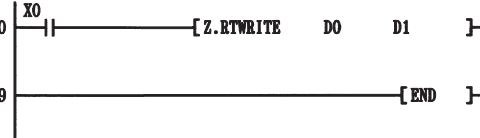
- (1) 在以下发生运行错误情况中，错误标志 (SM0) 变为 ON，并且错误代码存储在 SD0 中。
- n 指定的数据不在 1 到 239 的范围内 (错误代码: 4100)。

● 从Ⓢ开始的数据超过了单个设定范围 (错误代码: 4100)。

[程序示例]

(1) 当 X0 为 ON 时，以下程序将 D1 到 D3 指定的路由信息写到 D0 指定网络号的网络模块中。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	X0
1	Z.RTWRITE	D0 D1
9	END	

[操作]

D0	1
D1	20
D2	1
D3	3

[路由参数设定内容]

步数指令软元件	中继网络号	中继网络号	路由站号
1	20	1	3
2	10	2	2
3	10	1	2

9. QCPU 指令

下面是现有的 QCPU 指令：

(1) 所有 QCPU 现有的指令

分类	指令	含义	参考章节
读模块信息	UNIRD UNIRDP	● 从指定的 I/O 起始号中读取模块信息	9.1 节
追踪置位	TRACE	● 将追踪文件中的追踪数据存储到内存卡中	9.2 节
追踪复位	TRACER	● 用 TRACE 指令复位数据设置	
写入数据到指定文件	FWRITE	● 写入数据到指定的文件中	9.3 节
从指定的文件中读取数据	FREAD	● 从指定的文件中读取数据	9.4 节
从内存中装载程序	PLOADP	● 传输一个存储在内存卡或者标准内存（除了驱动器 0）的程序到驱动器 0，同时设置为备用状态	9.5 节
从程序内存中卸载程序	PUNLOADP	● 把存储在标准内存（驱动器 0）的备用程序删除	9.6 节
装载+卸载	PSWAPP	● 删除存储在标准内存（驱动器 0）的备用程序，然后传输一个存储在指定的内存卡或者标准内存（除了驱动器 0）的程序到驱动器 0，同时置于备用状态	9.7 节
文件寄存器的高速块传输	RBMov RBMovP	● 从被⑤指定的软元件批量传输 16 位数据的 n 点到从被⑥指定的软元件开始的 n 点区域	9.8 节

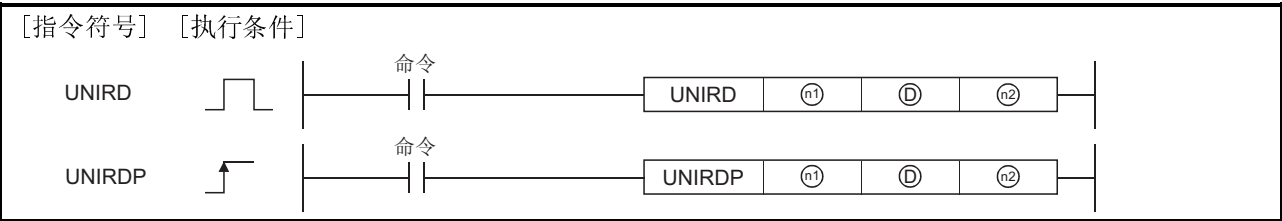
(2) 功能版本 B 增加的指令

分类	指令	含义	参考章节
写入到自站的 CPU 共享内存	S. TO	写自站的软元件数据到自站 CPU 模块的共享内存区域	9.9 节
从另一个站的 CPU 共享内存中读取	FROM	从另一个站的 CPU 共享模块或者运动控制器的共享内存区域读取软元件数据到自站	9.10 节
CPU 共享内存的自动刷新	COM	执行智能功能模块，一般数据处理的自动刷新和 CUP 共享内存的自动刷新	9.11 节

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
○	○	○	×	×

9.1 读取模块信息 (UNIRD(P))

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J [] []		智能功能模 块 U [] G []	变址 Zn	常量	
	位	字		位	字			K, H	\$
①	○	○	—	—	—	—	○	—	
②	—	○	—	—	—	—	—	—	
③	○	○	—	—	—	—	○	—	



[数据设定]

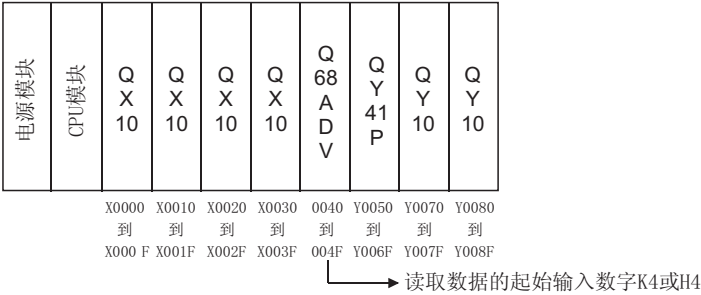
数据设定	含义	数据类型
Ⓜ	将从模块信息被读取的模块开始的起始 I/O 号整除 16 所获得的值 (0 到 FFH)	BIN 16 位
Ⓓ	储存模块信息的软元件起始号	软元件名称
Ⓝ	读取数据的点数 (0 到 256)	BIN 16 位

[功能]

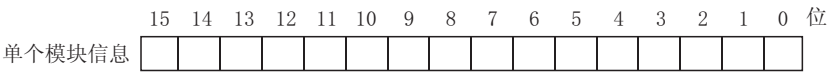
- (1) 从Ⓜ指定的模块中读取尽可能多的Ⓝ指定的模块信息，（通过将起始 I/O 号整除 16 获得该值），同时在软元件里存储该信息，然后由Ⓓ指定。
（读取实际安装模块的状态而不是通过 I/O 分配指定的模块类型。）

备注

- 1) 当它用 16 进制符号表达成 4 位数时，n1 的值由读取模块信息的插槽中的起始 I/O 号的前三位数来指定。



模块信息的细节描述如下：



位数	条目	含义	
0	I/O 点的数目	000: 16 点	001: 32 点
1		010: 48 点	011: 64 点
2		100: 128 点	101: 256 点
		110: 512 点	111: 1024 点
3	模块类型	000: 输入模块	001: 输出模块
4		010: I/O 混合模块	011: 智能功能模块
5			
6	外部电源状态（为将来扩充）	开：外部电源连接 关：外部电源没有连接	
7	有/没有熔断的保险丝	开：一些模块有熔断的保险丝 关：正常	
8	在线模块替换状态	开：执行在线模块替换 关：在线模块替换已经结束或没有执行在线模块替换	
9	轻微/中等错误状态	开：轻微/中等错误出现 关：正常	
10	模块错误状态	00: 没有模块错误	01: 轻微错误
11		10: 中度错误	11: 严重错误
12	模块备用状态	ON: 正常	OFF: 模块错误出现
13	空缺	—————	
14	A/Q 模块	开：A 一系列模块	关：Q 一系列模块
15	模块安装状态	开：模块已经安装	关：模块没有安装

[运行错误]

(1) 在以下出现运行错误的情况中，错误标志（SM0）变为 ON，并且错误代码存储在 SD0 中。

[高性能型 QCPU 和过程控制 CPU]

- 当n1不属于 0 到 FFH

(错误代码：4100)
- 当n2不属于 0 到 256

(错误代码：4100)
- W 当n1和n2等于或远大于 257

(错误代码：4100)

[Q00/Q01CPU]

- 当n1不属于 0 到 3FH

(错误代码：4100)
- 当n2不属于 0 到 64

(错误代码：4100)
- 当n1和n2等于或远大于 65

(错误代码：4100)

[Q00JCPU]

- 当n1不属于 0 到 FH

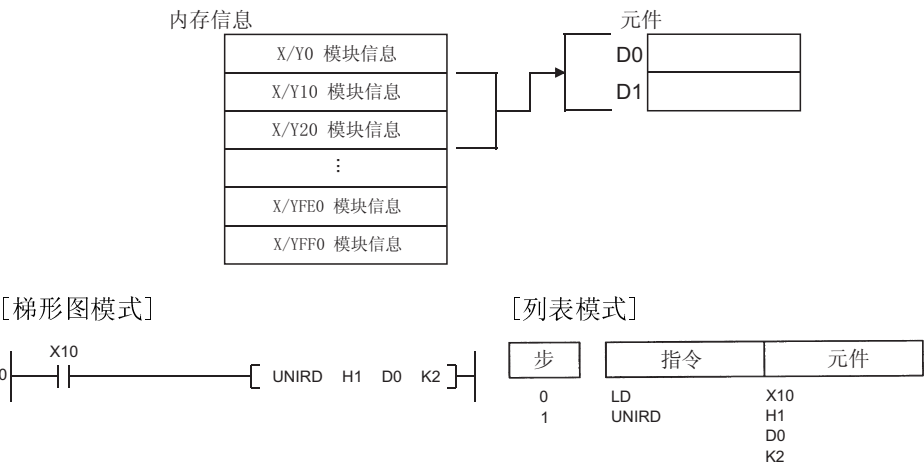
(错误代码：4100)
- 当n2不属于 0 到 16

(错误代码：4100)
- 当n1和n2等于或远大于 17

(错误代码：4100)

[程序示例]

(1) 当 X10 为 ON 时，下列程序从 D0 开始，在 I/O 号码 10H 到 20H 存储模块信息。



[梯形图模式]

0

X10

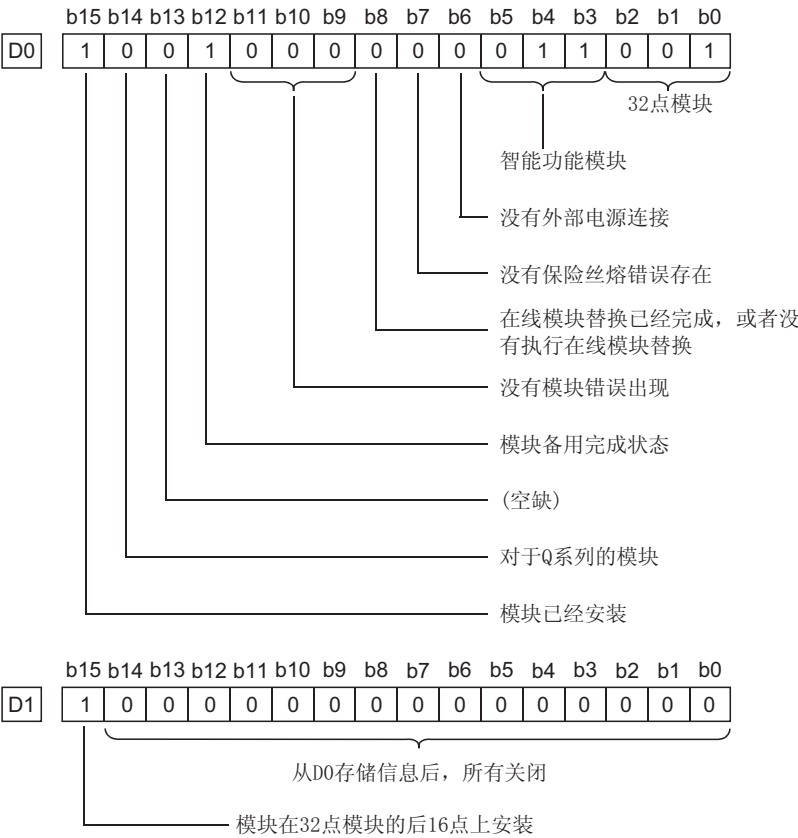
UNIRD H1 D0 K2

[列表模式]

步	指令	元件
0	LD	X10
1	UNIRD	H1 D0 K2

读出结果（当从 D0 读时）

① 对于 Q 系列的 32 点智能功能模块



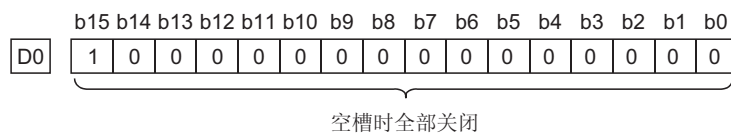
- 使用 48 或 64 点模块，D1 的相同内容被分别存储在 D2 或 D2 和 D3 之间。

② 对于 A 系列的 32 点模块

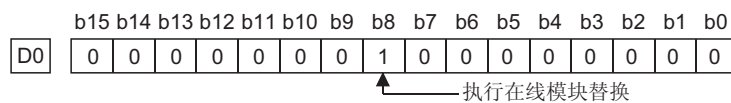


- 使用 48 或 64 点模块，D1 的相同内容被分别存储在 D2 或 D2 和 D3 之间。

③ 空槽



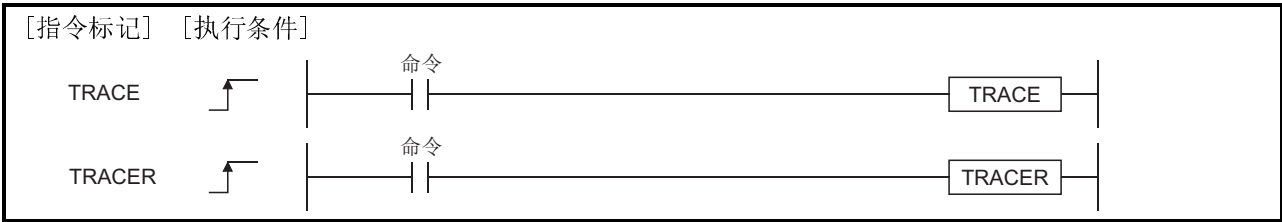
④ 执行在线模块替换



QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	×	×

9.2 追踪置位/复位（TRACE, TRACER）

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H)		特殊功能模 块 U[]G[]	变址寄存器 Zn	常量	其它
				直接 J [] []					
	位	字		位	字				
—	—								



[功能]

TRACE

- (1) 当 SM800, SM801 和 SM802 为 ON 时，TRACE 指令存储由外围软元件指令的追踪数据，通过指定数字，存储在内存卡的追踪文件里。当 TRACE 指令执行时，SM803 为 ON。
在 TRACE 指令执行后，采样通过采样追踪的指定数字进行重复，然后数据锁定，追踪结束。
- (2) 在追踪执行的过程中，如果 SM801 为 OFF, 取样结束。
- (3) 在 TRACE 指令执行和追踪结束后，SM805 为 ON。
- (4) 在 TRACE 指令执行和追踪结束后，SM805 为 ON。
当 TRACE 指令执行时，TRACE 指令能够再次执行。

TRACER

- (1) TRACER 指令复位 TRACE 指令。
- (2) 当 TRACER 指令执行后，SM803 到 SM805 为 OFF。
- (3) 当 TRACER 指令执行后，TRACE 指令能再次执行。

备注

- 1) 在使用追踪操作过程中，可以参考 CPU 模块的用户手册（功能解释，编程基础）。
- 2) 在使用外围设置执行追踪时，可以参考 GX Developer 的操作手册。

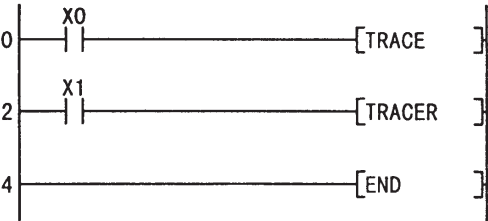
[运行错误]

(1) 没有与 TRACE 指令相关的运行错误。

[程序示例]

(1) 当 X0 为 ON 时，下列程序执行 TRACE 指令，当 X1 为 ON 时，可以用 TRACER 指令复位 TRACE 指令。

[梯形图模]



[列表模式]

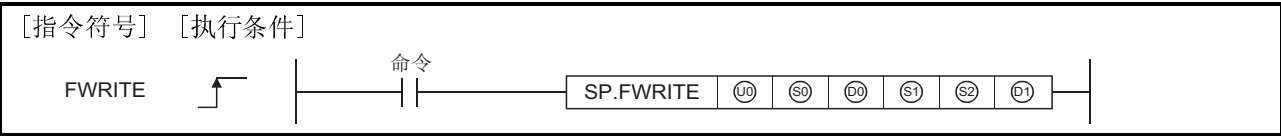
步	指令	软元件
0	LD	X0
1	TRACE	
2	LD	X1
3	TRACER	
4	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	×	×

9.3 写数据到指定的文件（FWRITE）

设定 数据	可用软元件									
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J [][]		特殊功能模 块 U [] G []	变址寄存器 Zn	常量		其它
	位	字		位	字			K, H	\$	
(S0)	○	○	—				○	—	—	
(D0)	—	○	—				—	—	—	
(S1)	—	○	—				—	—	—	
(S2)	—	○	—				—	○	—	
(D1)	△*	△*	—				—	—	—	

*:本地软元件和指定用于个人程序的软元件不能使用。



[设定数据/控制数据]

设定数据	含义			设定范围	设定者	数据类型
①	虚拟			—	—	BIN 16 位
②	软元件指定			2	用户	
③	软元件的起始号码存储控制数据，以下的控制数据是需要的。					
	软元件	条目	含义/设定数据	设定范围	设定者	
	④	执行/完成类型	指定执行类型 H0000 : 写入二进制数据 H0100 : 在 CSV 格式转换后写入数据	H0000 H0100	用户	
	⑤+1	(未使用)	由系统使用		系统	
	⑤+2	写入结果（已写数据的数目）	包含相对于②指定数据的实际写入数据的数目。数值单元由字/字节的单元指定来确定。		系统	
	⑤+3	(未使用)				
	⑤+4 ⑤+5	文件中的位置	当二进制数据写入由④指定时，指定文件中开始写入的位置 H00000000 : 从文件开端 H00000001 到 HFFFFFFFE : 从指定的位置开始。 值的单位由字/字节的单元指定来确定。 HFFFFFFF : 添加到文件尾部 CSV 格式写入指定在⑥下 对于高性能型 QCPU，系列号的前五位数字是 01111 或更小时，通常要设置文件的开始(0h)。 对于高性能型 QCPU/过程控制 CPU，系列号的前五位数字是 01112 或更大，设置文件的位置。 H00000000 到 HFFFFFFF : 在文件的开端开始 HFFFFFFF : 在文件的末尾添加	H00000000 到 HFFFFFFF	用户	
	⑤+6	列指定的号码	当二进制写入指定在⑥的情况下，一直设置为 0，当 CSV 格式写入指定在⑥的时候，设置列数字，数据写为 0 : 设置每行的列数 除了 0 以外 : 设置指定的列号	0 到 65535	用户	

[设定数据/控制数据] (继续)

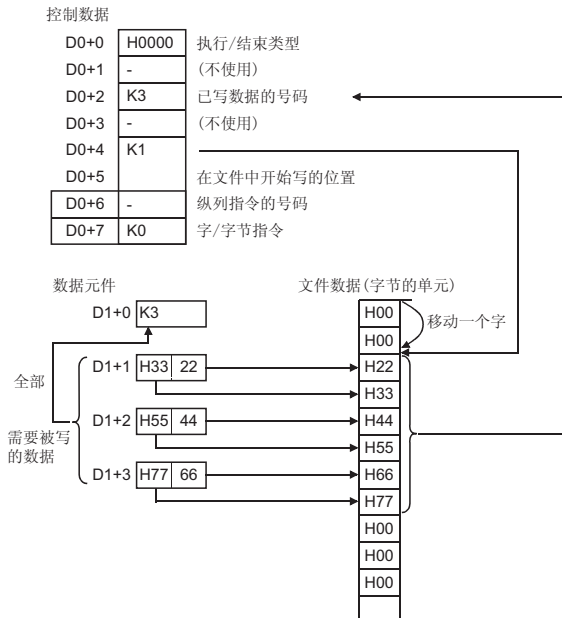
设定数据	含义			设定范围	设定者	数据类型	
①①	①①+7	字/字节指定	0: 字 1: 节	0, 1	用户	BIN 16 位	
存储一个文件名称的软元件起始号, 文件名称表达如下:							
	软元件	条目	含义/数据设定		设定范围		设定者
①②	①②+1 到 ①②+□	文件名称字符串	指定一个文件名称的字符串 ● 当省去扩展名时, 也省掉 “.” (句点) ● 限制文件名到 8 个字符+句点+3 个字符。 ● 当有 9 个或者更多字符使用时, 不论扩展名存在与否将被省略。扩展名 BIN 或 CSV 将自动分配。		字符串		用户
存储数据的软元件的起始号, 已写入数据表达如下							
	软元件	条目	含义/数据设置		设定范围	设定者	
①③	①③	被写入数据的号码	指定数据的数字要求写入 (用字的单元) 尽管字节由①①指令, 数据也应当用字的单元来指令。		1 到 480	用户	
①③+1 到 ①③+□		将被写入的数据	需要被写的数字		H0000 到 HFFFF		
位软元件在指令完成时为 ON。(①④+1 也可以在错误结束时继续为 ON)							
	软元件	条目	含义/设定数据		设定范围	设定者	
①④	①④	结束信号	指示指令的完成 开: 结束 关: 没有结束			系统	
①④+1		错误结束信号	表明指令是正常结束或者是错误结束 开: 错误结束 关: 正常结束				

备注

- (1) 在 S0 (驱动指定), 你只可以设置 ATA 卡驱动 (2)
注意当 FLASH 卡装载时, FWRITE 指令不能执行写入的功能。
你不能设置 SRAM 卡, 标准 RAM 或标准 ROM 驱动。
- (2) 对于 CSV 设置, 已写数据是十进制值。
例如) 字符 “A” (41H) — “65” 被写入
处理范围: -32768 到 32767
- (3) 对于二进制写入, 指定字文件位置的设置范围是从 H00000DA0 到 H7FFFFFFF 和 HFFFFFFF。

[功能]

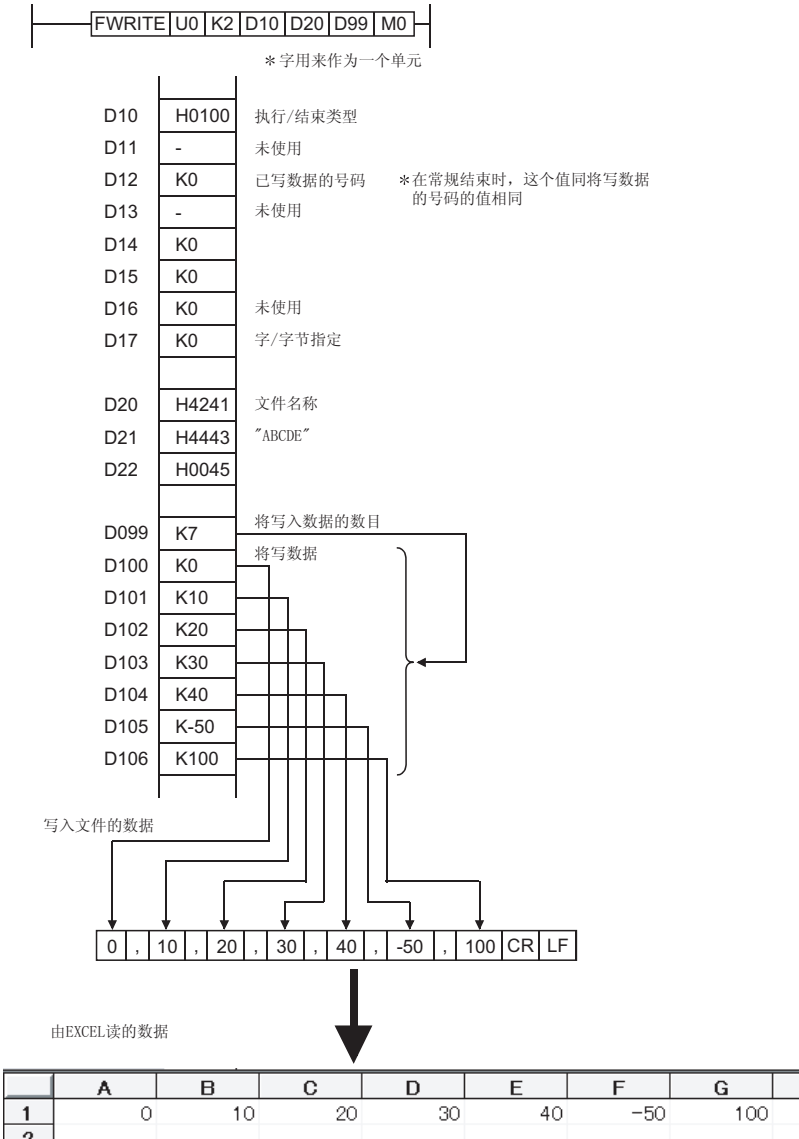
- (1) 指定的数据数目写入到指定的文件。在控制数据中设置执行/结束类型，从而指定是否不通过任何转换写入二进制数据或者在写入之前转换二进制数据成 CSV 格式数据。
(写入目标只有 ATA 卡)
- (2) 在检测到 FWRITE 的指令结束和 END 指令执行后，结束信号位软元件 (E1+0) 自动为 ON。在结束指令执行后开始下一个扫描时，位软元件为 OFF。
对于 FWRITE 指令使用位软元件作为执行结束的标志。
当 FWRITE 指令异常结束时，错误结束软元件 (E2+1) 与执行结束软元件 (E2+0) 同步 ON/OFF。对于 FWRITE 指令使用这个软元件作为出现错误结束的标志。
SM721 在指令执行时是为 ON。
当 SM721 为 ON 时，FWRITE 指令不能执行。(如果尝试，没有执行处理)
当一个错误在指令执行之前被检测到 (在 SM721 为 ON 之前)，执行结束软元件 (E2+0)，错误结束软元件 (E2+1) 和 SM721 不变为 ON。
- (3) 确信使用字的单元去指定被写入数据的数字 (S+0) 和文件中的位置 (E9+4 和 5)。如下实例表明如何处理单个软元件数据的二进制数据写入操作：



- (4) 当写入二进制数据时
- ① 如果目标文件的扩展名是省略的，使用“BIN”作为扩展名。
 - ② 当指定文件不存在时，将建立一个新的文件，添加数据，同时从文件的开始部分存储。
新文件的属性使用档案属性来设定。
 - ③ 当存储数据的大小超过写入的过程中文件里面存在的区域，多余的数据将被添加/存储。
 - ④ 如果指定的文件位置大于存在文件的尺寸。
 - 高性能型 QCPU 的系列号的前五位数字是“01111”或更小将会出现错误。
 - 高性能型 QCPU/过程控制 CPU 的系列号的前五位数字是“01112”或更大将在 0 点时执行写入同时正常地结束。
 - ⑤ 当数据添加/存储，缺乏足够的空间时，错误将出现。在这种情况下，成功添加/存储的数据仍然保留在中间存储器中。在尽可能多的数据被添加/存储后，错误结束将指示。

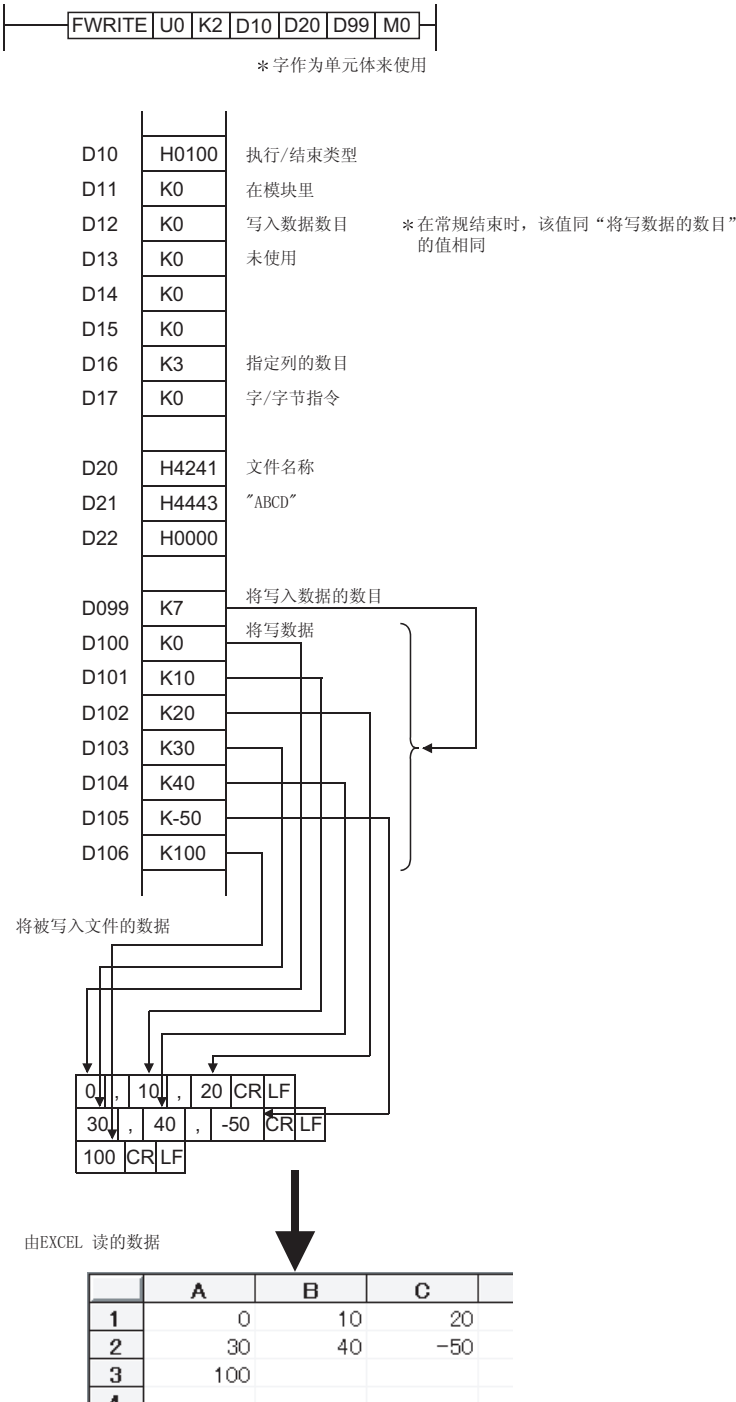
- (5) 在 CSV 格式转换后写入数据
- ① 如果扩展名省略，可以使用“CSV”作为扩展名。
 - ② 当存在的文件被指定时
 - [系列号的前 5 位数是“01111”或更早的高性能型 QCPU]
文件内容将被全部删除同时数据从开始处存储。
 - [系列号的前 5 位数是“01112”或更晚的高性能型 QCPU/过程控制 CPU]
 - 当不是 HFFFFFF 设置在 (D0+4, D0+5) 时，文件内容将被全部删除同时数据从开始处存储。
 - 当 HFFFFFF 设置在 (D0+4, D0+5) 时，数据将从文件的末尾开始存储。
 - ③ 当指定文件不存在时，将建立一个新的文件，数据从文件的开始部分添加/存储。
新文件的属性使用档案属性来设定。
 - ④ 当数据添加/存储，缺乏足够的空间时，错误将出现。在这种情况下，成功添加/存储的数据仍然保持在中间存储器中。在尽可能多的数据被添加/存储后，错误结束将指示
 - ⑤ 当列的指定列数是“0”时，数据作为单行数据存储在 CSV 格式文件中。
以下是这种情况的一个实例：

当数据在 CSV 格式转换后写入时，同时指定的列数是不为“0”



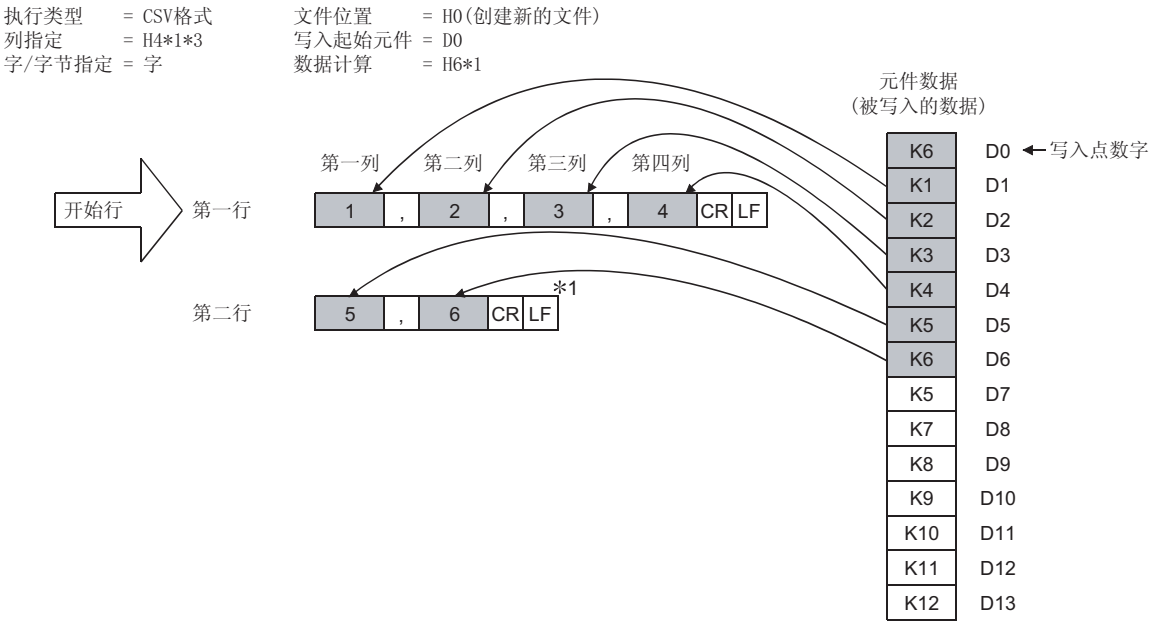
- ⑥ 当数据在 CSV 格式转换后写入，同时指定的列数不是“0”，数据作为表格数据以指定的列数存储在 CSV 模格式的文件里面。
 以下是该情况的一个实例：

当数据在 CSV 格式转换后写入，同时列的指定号码不是“0”。

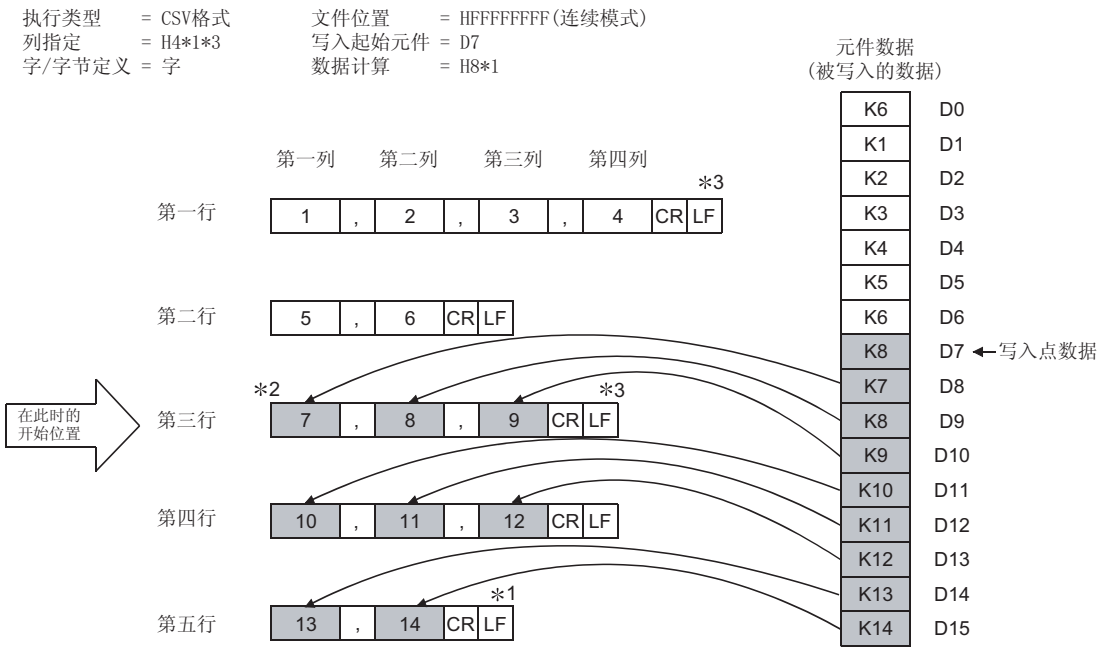


当通过高性能型 QCPU/过程控制 CPU 添加数据时，系列号的前五位数是“00012”或更大

[指定文件到数据被写入的位置] (如果一个文件存在，删除它并重新创建一个文件。)



[在增加模块中，在文件的末尾处添加]



- *1: 除非写入点的数字设定成“列指定”的整倍数，列的数目是随机的。
- *2: 由于最后的数据总是紧随行供给代码，添加通常在添加模式中开始于新一行的开始部分。
- *3: 在添加模式中，如果“列指定”从先前写入的模式中改变，列数将改变。

⑦ 在中断程序中不要执行 FWRITE 指令。
(操作结果不确保)

[运行错误]

- (1) 在下列出现运行错误情况中，SM0 变为 ON, 并且错误代码存储在 SD0 中。
- 通过驱动指定软元件⑤指定的驱动包含了除 ATA 卡外的中间存储器

(错误代码: 4100)

● 在控制数据⑥中指定的值和其后软元件超出设置范围。

(错误代码: 4100)

● 通过“写入数据的数目”(⑤+0)指定的值超出设置范围，或超过由(⑤+1)或其后软元件指定的软元件范围

(错误代码: 4101)

● 中间存储器中的空余空间不足

(错误代码: 4100)

● N 在尝试建立一个新的文件时，不能找到多余的空间

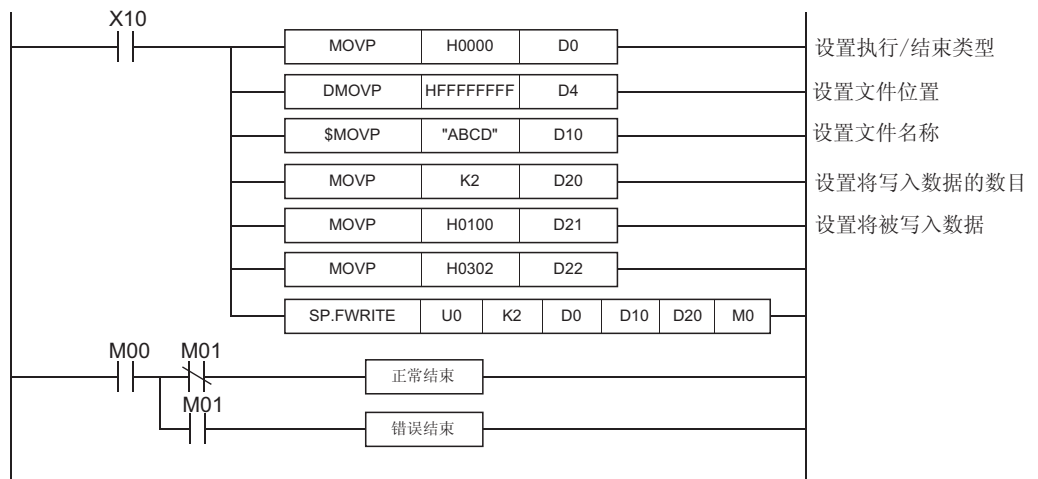
(错误代码: 4100)

● 指定无效的软元件

(错误代码: 4004)

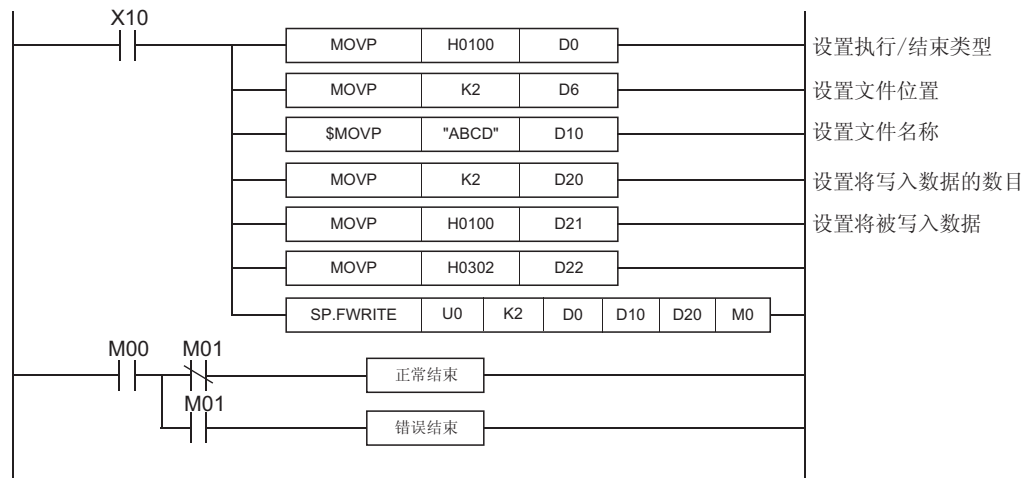
[程序示例]

- (1) 当 X10 为 ON 时，下列程序添加二进制数据 (H00, H01, H02, H03) 的 4 个字节到插入的驱动 2 的内存卡的文件“ABCD.BIN”中。
- 假设来自⑥的 8 个点为控制数据软元件保留



(2) 当 X10 为 ON 时，下列程序在驱动 1 的嵌入内存卡中建立一个新的文件“ABCD.CSV”，同时在 CSV 格式中写入数据的四个字节作为两列表格数据。

- 假设来自⑥的 8 个点保留给控制数据软元件。



- 写入文件如下表示：

0	,	1	,	CR	LF
2	,	3	,	CR	LF

被写入文件的内容

↓ 由EXCEL读取的数据

	A	B
1	0	1
2	2	3

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	×	×

9.4 从指定文件中读取数据（FREAD）

设定 数据	可用软元件									
	内部软元件		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G]	变址寄存器 Zn	常量		其它
	位	字		位	字			K, H	\$	
(S0)	○	○	—				○	—	—	
(D0)	—	○	—				—	—	—	
(S1)	—	○	—				—	—	—	
(D1)	—	○	—				—	○	—	
(D2)	△*	△*	—				—	—	—	

*:本地软元件和指定用于个人程序的软元件不能使用。



[设定数据/控制数据]

设定数据	含义			设定范围	设定者	数据类型
④	虚拟			0 到 H00FF	用户	BIN 16 位
⑤	软元件指定			2	用户	
⑥	存储控制数据的软元件的起始号 需要下列控制数据					
	软元件	条目	含义	设定范围	设定者	
	⑥	执行/结束类型	指定执行类型 H0000 : 读取二进制数据 H0100 : 读在 CSV 格式转换后	H0000 H0100	用户	
	⑥+1	(未使用)	被系统使用		系统	
	⑥+2	被读数据号	指定需要读取数据的号码(单位: 字) 然字节由⑥+7 指定时, 该数据也应当以字的单位来指定。	1 到 480	用户	
	⑥+3	(未使用)				
	⑥+4 ⑥+5	文件中的位置	当二进制数据读取由⑥指定时, 在文件中指定开始的位置 H00000000 : 从文件开始 H00000001 到 HFFFFFFFE : 从指定的位置开始。 值的单位由字/字节的单元指定来决定 HFFFFFFF: 不能设置 CSV 格式写入指定在⑥下 对于高性能型 QCPU, 系列号的前五位数字是 01111 或更小, 通常要设置文件的开始位置(0h)。 对于高性能型 QCPU/过程控制 CPU, 系列号的前五位数字是 01112 或更大, 设置文件的位置。 H00000000 : 从文件的开始位置读取数据 H00000000 到 HFFFFFFF : 从指定的行开始读取数据 HFFFFFFF : 从先前读取的位置开始, 继续读取数据	H00000000 到 HFFFFFFFE / HFFFFFFF	用户	
	⑥+6	列数指定	二进制写入数据时规定在⑥的情况下, 一直设置为 0, CSV 格式写入数据时指定在在⑥下, 从要读取数据的地方设置列的数目 0 : 没有列, 认为只有一行 除了 0 : 认为是指定的列数	0, 1 到 65535	用户	
	⑥+7	字/字节指定	0: 字 1: 字节	0, 1	用户	
⑦	存储文件名称的软元件的起始号。文件名如下:					
	软元件	条目	含义/设定数据	设定范围	设定者	
	⑦ 到 ⑦+□	文件名称 字符串	指定文件名的字符串 ● 当省略扩展名时, 同样省略“.”(句点) ● 限定文件名称最多由 8 个字符串+句点+3 个字符串 ● 当超过 9 个字符串被使用时, 扩展名不论出现与否将被省略。扩展名“BIN”“CSV”自动指派。	字符串	用户	

[设定数据 /控制数据] (续)

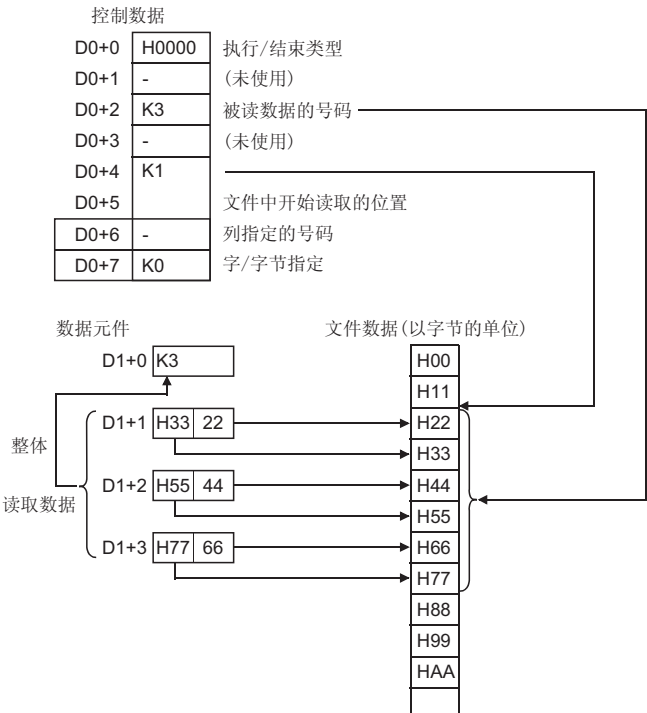
设定数据	含义			设定范围	设定者	数据类型
D1	用于存储读数据的软元件起始号。					
	软元件	条目	含义/数据设置	设定范围	设定者	
	D1	读取结果（读数据的数目）	相对于由D0+2 指定的数据，包含了实际读取的数据单元的值由字/字节的单位定义来决定。	0 到 480	系统	
	D1+1 到 D1+□	将被读的数据	需要读取的数据	H0000 到 HFFFF	系统	
D2	二进制软元件在指令完成时继续。D2+1 也可以在错误结束时继续运行。					位
	软元件	条目	含义/数据设置	设定范围	设定者	
	D2	结束信号	指示指令的完成 开：结束 关：没有结束		系统	
	D2+1	错误结束信号	表明指令是正常结束或者是错误结束 开：错误结束 关：正常结束			

备注

- (1) 在Ⓔ0(软元件指定), 你只可以设置 ATA 卡驱动
注意当 FLASH 卡装载时, FWRITE 指令不能执行读的功能。
你不能设置 SRAM 卡, 标准 RAM 或标准 ROM 驱动。
- (2) 对于 CSV 设置, 写入的数据是十进制值。
例如) 字符 “A” (41_H) → “65” 被写入。
处理范围: -32768 到 32767
- (3) 在进行二进制读时, 指定字文件位置的设置范围是从 H00000DA0 H7FFFFFFF 和 HFFFFFFF。

[功能]

- (1) 从指定文件中读取数据
设置在控制数据中的执行/结束类型，从而指令是不通过任何转换读二进制数据或者在读之前转换二进制数据成 CSV 格式数据。（读取目标只有 ATA 卡。）
- (2) 在检测到 FREAD 的结束指令和 END 指令执行后，结束信号位软元件 (Ⓔ2+0) 自动为 ON。在结束指令执行后开始下一个扫描时，位软元件为 OFF。
使用位软元件作为 FREAD 指令执行结束的标识。
当 FREAD 指令异常完成时，错误执行软元件 (Ⓔ2+1) 与执行结束软元件 (Ⓔ2+0) 同时 ON/OFF。使用这个软元件可以作为 FREAD 指令出现错误结束的标识。
SM721 在指令执行时是为 ON 的。
当 SM721 为 ON 时，FREAD 指令不能执行。（如果进行尝试，没有任何程序执行。）
当一个错误在指令执行之前被检测到（在 SM721 为 ON 之前），执行结束软元件 (Ⓔ2+0)，错误结束软元件 (Ⓔ2+1) 和 SM721 不变为 ON。
- (3) 确信使用字里面的单元去指定被读的数据号码 (Ⓔ0+2) 和文件的位置 (Ⓔ0+4 和 5) 以及读数据号 (Ⓔ3+0)。
如下实例表明单个软元件数是如何处理成二进制数据的操作。



- (4) 当读取二进制数据时
 - ① 如果目标文件的扩展名时省略的，使用“BIN”作为扩展名。
 - ② 如果指定文件不存在，一个错误将出现
 - ③ 如果指定的位置大于存在文件的大小
 - 高性能的 QCPU 模型的系列号的前五位数字是“01111”或更早将会出现错误。
 - 高性能的 QCPU 模型的系列号的前五位数字是“01112”或更迟将在 0 点时执行写入同时正常的结束
- (5) 在 CSV 格式转换后读取数据
 - ① CSV 格式文件的元素按每行读取 (EXCEL 的单元)。数量值和字符串被转化成二进制数据存储在软元件里面。
 - ② 如果扩展名省略，“.CSV”作为扩展名。
 - ③ 当指定文件不存在时，错误将出现。

- ④ 文件开始时的元素通过由被读数据号码 (ⓐ+2) 指定的数字而读取。
当文件最后的数据在数据的指定数字读取之前得到时
- 系列号的前 5 位数是“01111”或更小的高性能型 QCPU 将导致错误。
 - 系列号的前 5 位数是“01112”或更大的高性能型 QCPU/过程控制 CPU 将在点 0 处执行写操作，并能正常完成。

- ⑤ 当指定的列数是” 0” 时,数据在 CSV 格式文件里通过忽略行来读取。
下面是一个该情况的实例:

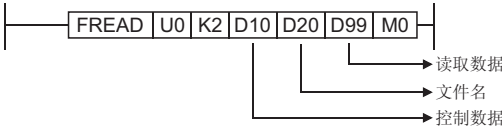
数据由EXCEL 创建

	A	B	C
1	主要/分项		测量值
2	长度	1	3
3	温度	-21	

存储在CSV格式里的数据

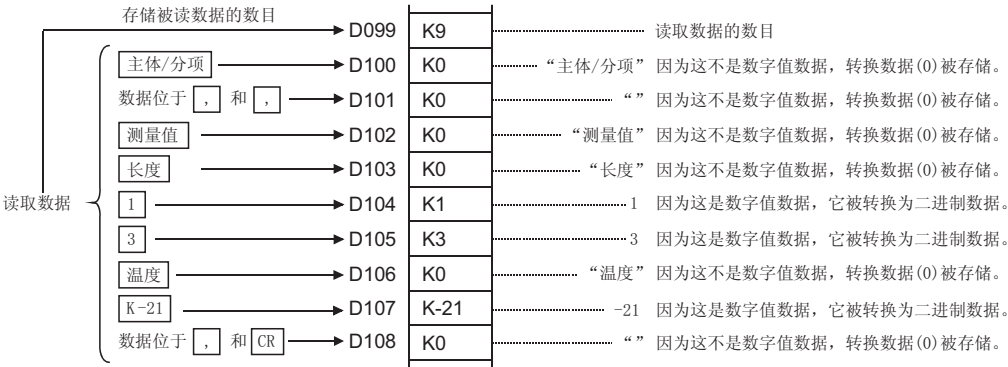
主体/分项	,	,	测量值	CR	LF
长度	,	1	, 3	CR	LF
温度	,	-21	,	CR	LF

读到元件的数据

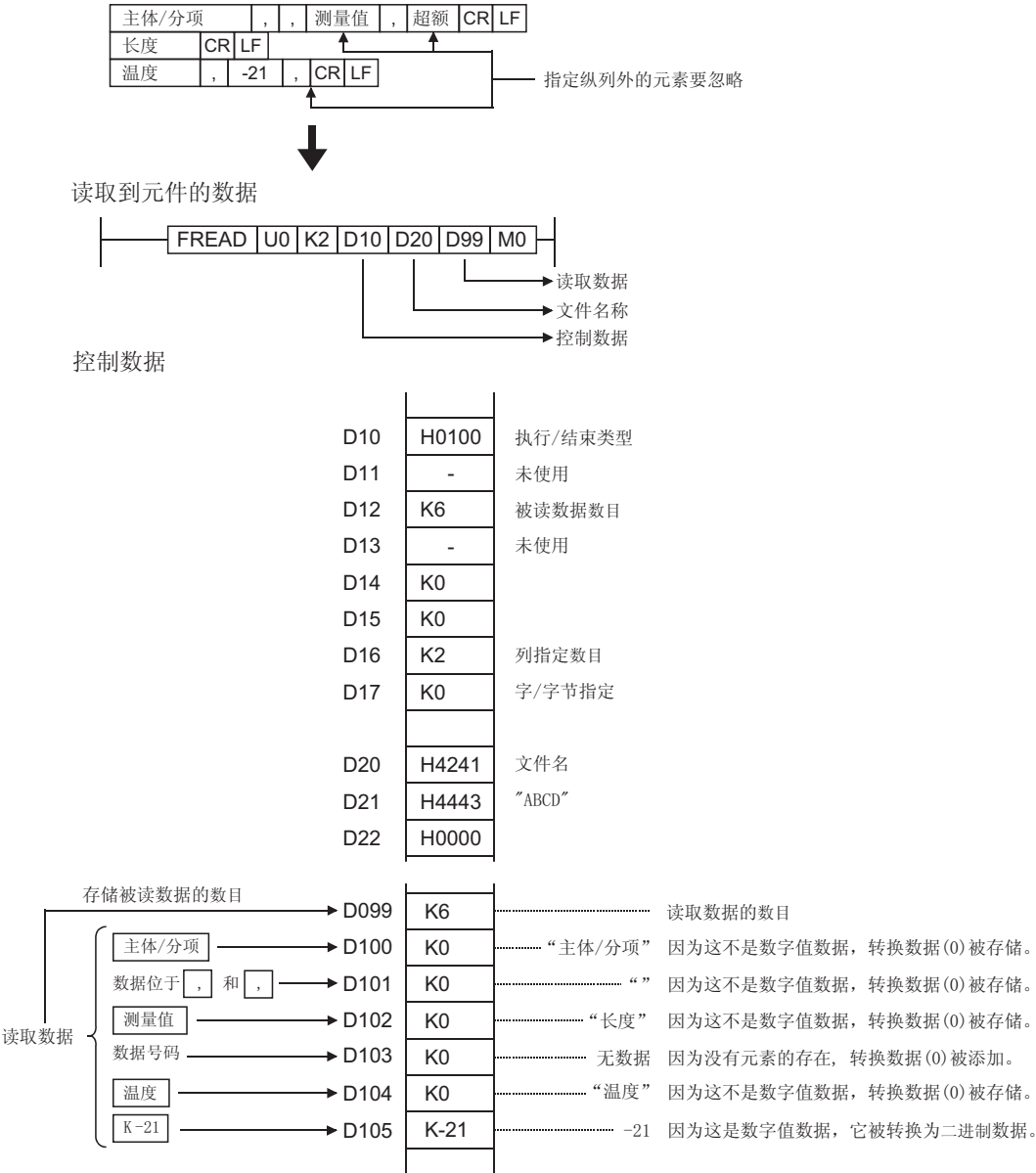


控制数据

D10	H0100	执行/结束类型
D11	-	未使用
D12	K9	被读数据数目
D13	-	未使用
D14	K0	
D15	K0	
D16	K0	列指定数目
D17	K0	字/字节指定
D20	H4241	文件名
D21	H4443	"ABCDE"
D22	H0045	



如果列数在每行都有变化,数据同样可以忽略行来读取。
(EXCEL 不能创建这样的文件。当一个用户修改 CSV 文件后才会发生)



- ⑥ 当数据在 CSV 格式转换后读取，同时指定的列数不是“0”，数据作为表格数据同指定的列数存储在 CSV 格式的文件里面。位于指定列外的元素需要忽略。
- 以下是该情况的一个实例：

由EXCEL创建的数据

	A	B	C
1	主体/分项		测量值
2	长度	1	3
3	温度	-21	



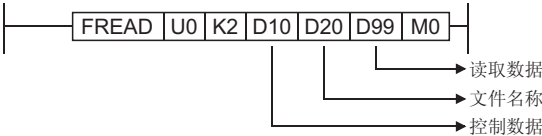
在CSV格式里存储的数据

主体/分项	,		,	测量值	CR	LF
长度	,	1	,	3	CR	LF
温度	,	-21			CR	LF

位于指定纵列外的元素被忽略

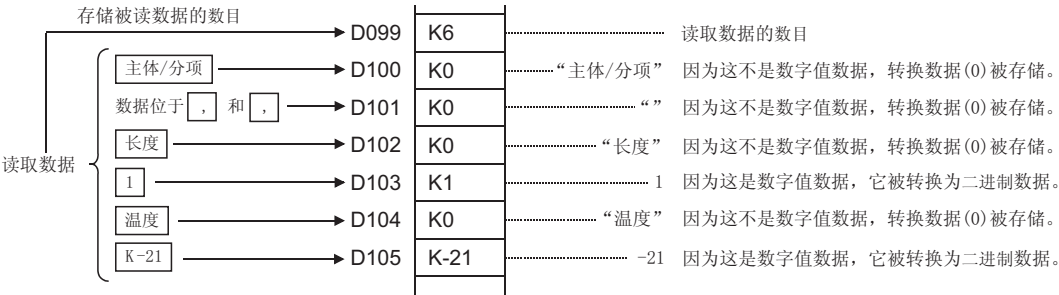


读取到元件的数据

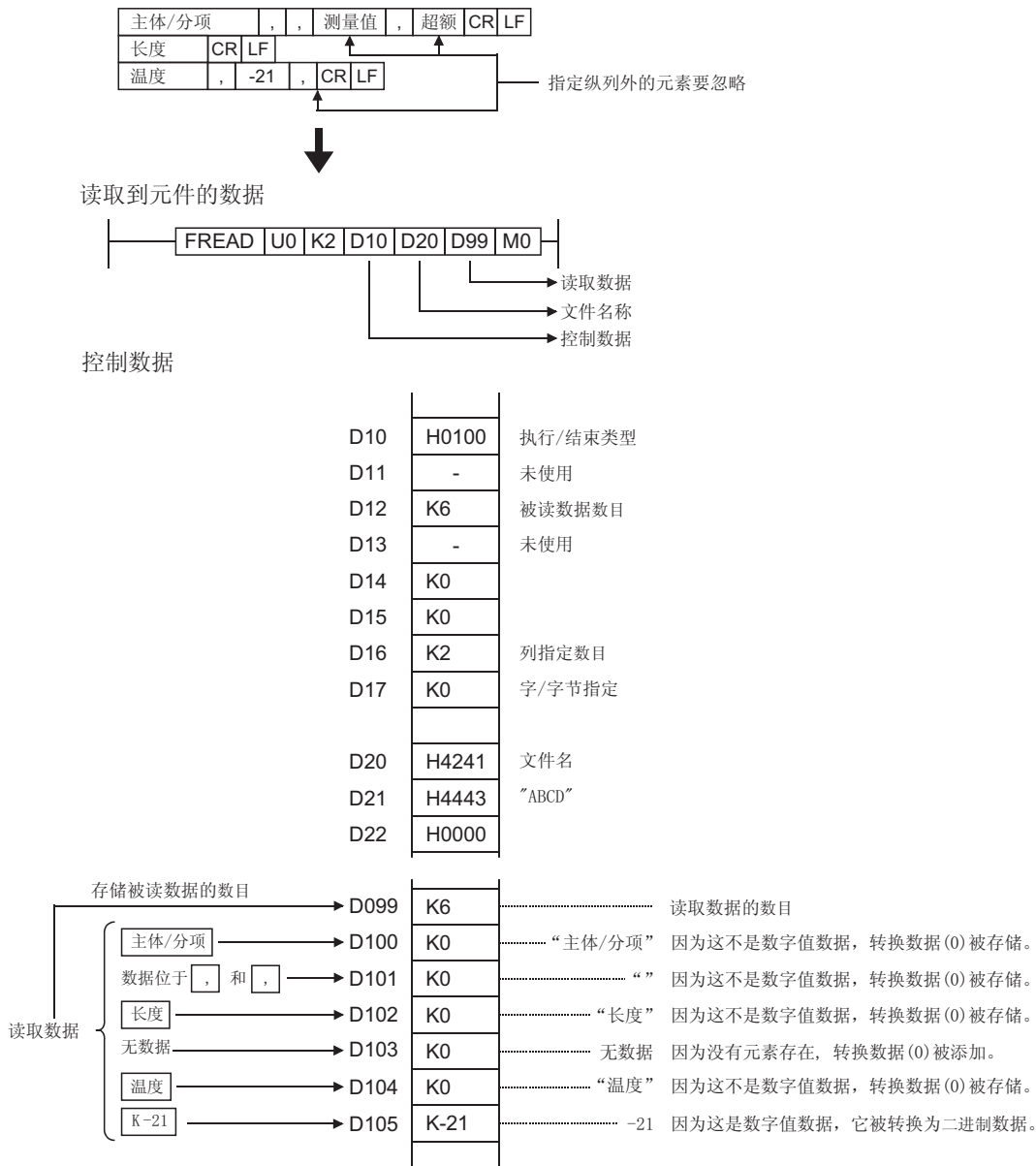


控制数据

D10	H0100	执行/结束类型
D11	-	未使用
D12	K6	被读数据数目
D13	-	未使用
D14	K0	
D15	K0	
D16	K2	列指定数目
D17	K0	字/字节指定
D20	H4241	文件名
D21	H4443	"ABCD"
D22	H0000	

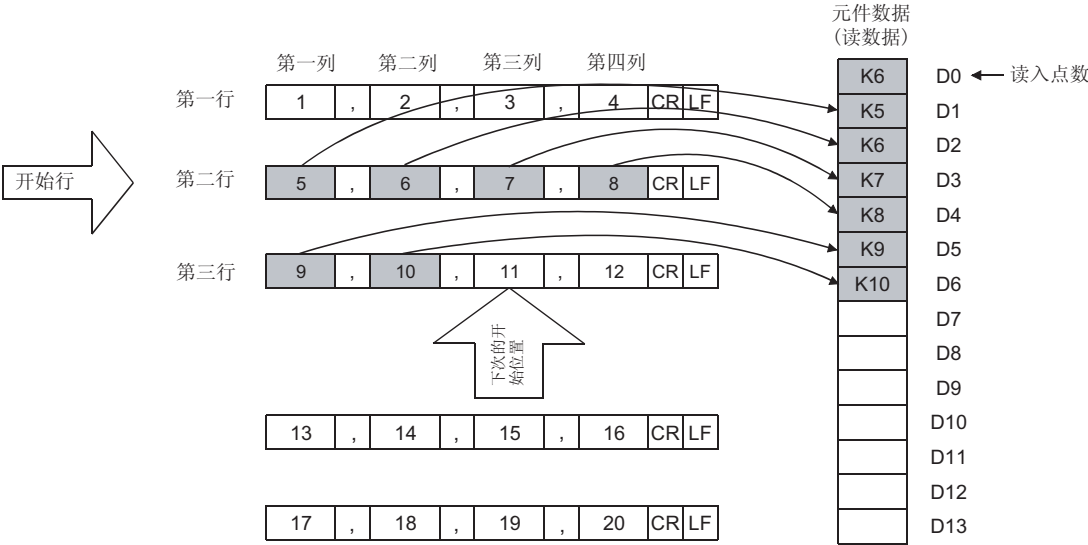


如果每行的列数都有变化,指定列外的元素将被忽略,同时“0”被添加到元素不存在的地方。



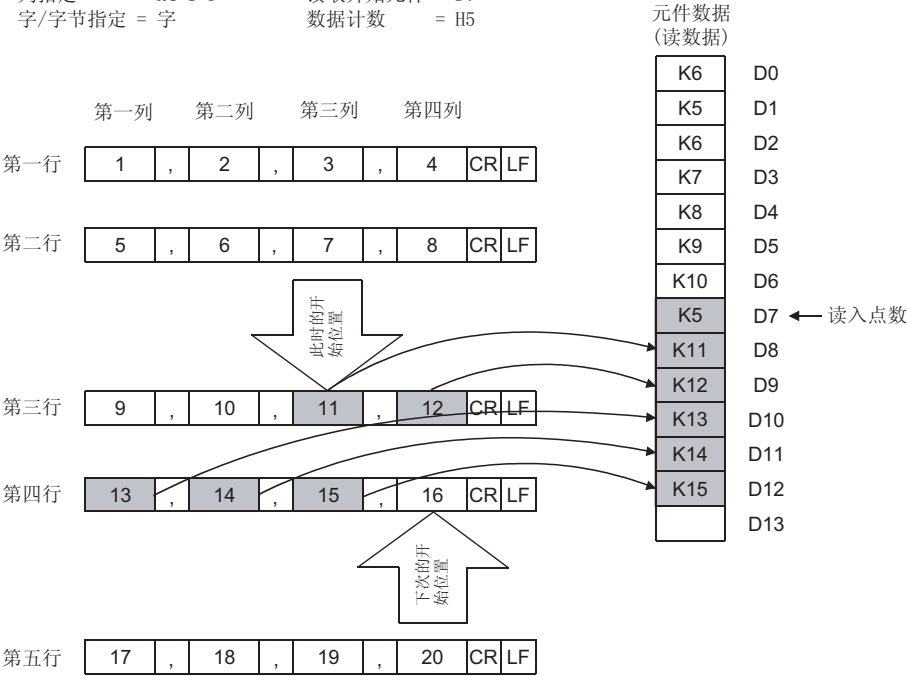
当数据被系列号的前五位是“01112”或更大的高性能型 QCPU/过程控制 CPU 多次整除或读取时
[指定你要开始读取的行]

执行类型 = CSV格式 开始行数 = H2
列指定 = H4*1*3 读入起始元件 = D0
字/字节指定 = 字 数据计数 = H6



[在连续模式里, 从先前读取位置的末尾处继续读取]

执行类型 = CSV模式 开始行数 = HFFFFFFF (连续模式)
列指定 = H4*1*3 读取开始元件 = D7
字/字节指定 = 字 数据计数 = H5



- 在持续模式中执行数据读取时, 如果“执行类型”, “列指定”和“字/字节指定”的设定不同与先前的情况, 此前的添加不能正常执行。
- 如果在另一设定中的 FREAD 指令和 FWRITE 指令在持续模式里数据被持续读取的时候进行执行, 先前的添加不能正常工作。

⑦ 当数据在 CSV 格式转换后读取时, 超过范围的数字值和在目标 CSV 格式文件里的除了数字值的元素将被转换为“0H”。

⑧ 当数据在 CSV 格式转换后读取时, 数字值按如下方式读取和转换:

在 CSV 格式里的数字值	字软元件	
	没有信号	有信号
-32768	32768	-32768
-1	65535	-1
0	0	0
1	1	1
32767	32767	32767
32768	32768	-32768
65535	65535	-1

⑨ 在中断程序中不能执行 FREAD 指令
(在中断程序中不能执行 FREAD 指令)

[运行错误]

- (1) 在下列出现运行错误情况中, SD0 变为 ON, 并且错误代码存储在 SD0 中。
- 通过驱动指定软元件 (Ⓔ) 指定的驱动包含除了 ATA 卡外的中间存储器
 - 在控制数据 (Ⓔ) 中指定的值和随后的软元件超出设置范围
 - 通过“被读数据的号码” (Ⓔ+2) 指定的值超出设置范围
 - 指派无效的软元件
 - 由文件名称字符串 (Ⓔ) 指定的文件名称或其后软元件不在指定的驱动器内。

(错误代码: 4100)

(错误代码: 4100)

(错误代码: 4101)

(错误代码: 4004)

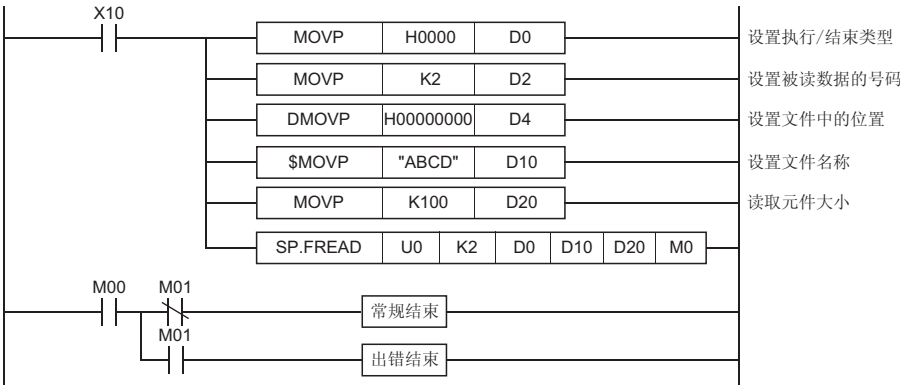
(错误代码: 2410)
- 读取数据的大小超过读取软元件的大小
 - 当二进制数据被读取时, 文件中数据的数字小于被读数据需求的数字指令的大小 (Ⓔ+2)。(高性能型 QCPU 的系列号的前五位数字是“01111”或更早)

(错误代码: 4101)

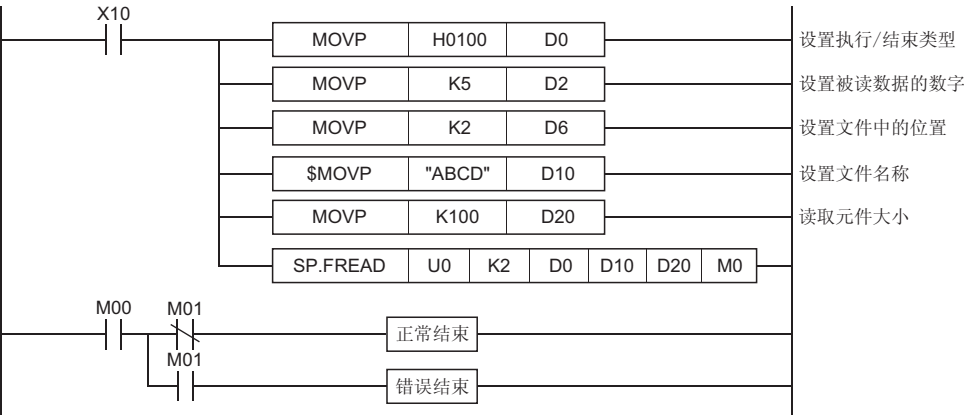
(错误代码: 4100)

[程序示例]

- (1) 当 X10 为 ON 时, 下列程序从嵌入到驱动器 2 的内存卡的文件“ABCD.BIN”的开始处读取二进制数据的四个字节。
- 假设来自 Ⓔ 的 8 个数据点为控制数据软元件保留
 - 假设来自 D20 的 100 个字节为读取软元件保留



- (2) 当 X10 为 ON 时, 下列程序从嵌入到插槽 0 中的内存卡中以 CSV 格式读取文件“ABCD.CSV”，作为两列表格数据。
- 假设来自 D0 的 8 个数据点为控制数据软元件保留
 - 假设来自 D20 的 100 字节为读取软元件保留
 - 假设目标 CSV 格式文件只包括数字值

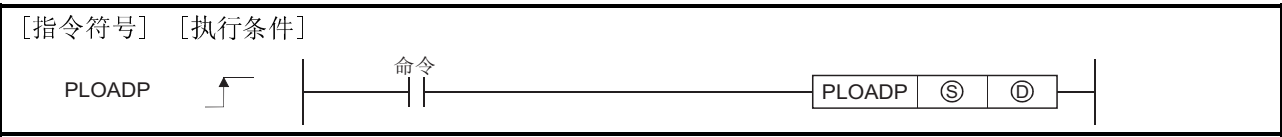


QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	×	×

9.5 从内存卡中装载程序(PLOADP)

设定数据	可用软元件						
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn
	位	字		位	字		
⑤	—	○		—			—
⑥	△*	—		—			—

※: 本地软元件不可使用。

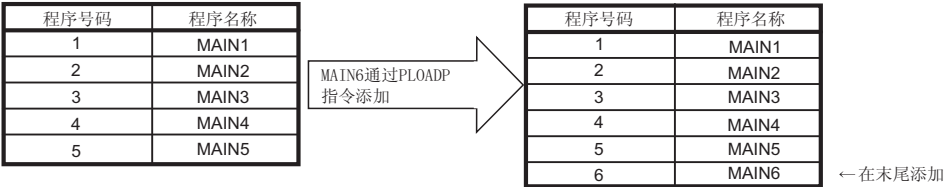


[设定数据]

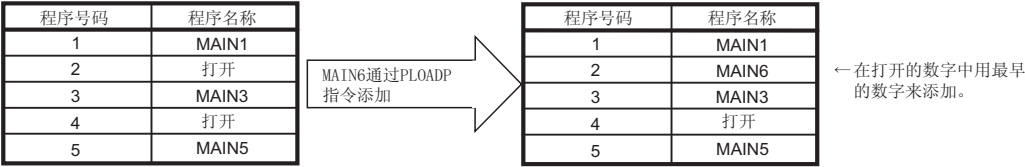
设定数据	含义	数据类型
⑤	存储要装载程序的驱动号码, 文件名的字符串数据, 或者存储字符串数据的软元件的起始号	BIN 16 位
⑥	指令结束时, 软元件变为 ON 一个扫描周期	位

[功能]

- (1) 存储在内存卡或标准 ROM 的程序将转移到程序内存。(驱动 0)
如果转移的文件没有注册到 PLC 参数对话框的程序设定里, 在 CPU 模块里的设定的程序将设为待命类型。
在这时, PLC 参数对话框的设定不再变化。(用 PLOADP 指令转移一个程序, 再程序内存中需要一个持续的空余空间。)
- (2) 使用 POADP 指令添加的的程序在未使用的程序号码中最早分配。
下面的例子假设 MAIN6 通过 PLOADP 指令来添加。
(a) 当程序号码连续的被设定, 新的程序在现在程序号码的末尾进行添加。
当 1 和 5 号程序已被设定, 新的程序添加作为程序 6。



- (b) 当有开放的程序号码时, 新的程序在为 ON 程序号码中用最早的号码添加。
(开放的程序号码在程序使用 PUNLOADP 指令删除时产生。)
当程序 2 和 4 为 ON 时, 新的程序添加作为程序 2。



- (3) 驱动 1, 2 和 4 可以指定 (驱动 3 不可以指定。)
- 驱动 1: 内存卡 (RAM)
 - 驱动 1: 内存卡 (RAM)
 - 驱动 4: 标准 ROM
- (4) 对于文件名称扩展名 (.QPG) 不需要指定。
- (5) 在指令结束时的 END 处理过程中, 由⑤指定的位软元件变为 ON。在下一个 END 程序处理的时候, 位软元件变为 OFF。
- (6) LOADP、PUNLOADP 和 PSWAPP 指令不能同时执行。如果两个或更多的上述指令同时执行, 随后需要执行的指令不能工作。
当使用上述指令时, 以用户来提供互锁避免同时执行。
- (7) 在中断程序时不能执行这个指令。
(在中断程序里执行这个指令可以导致功能故障)
- (8) 在执行使用 PLOADP 指令转移到程序内存的文件时, 使用 PSCAN 指令执行一个扫描执行类型。
(见章节 7.17.3)
- (9) 装载文件的 PLC 文件设定如下:
- ① 所有程序的文件用途
- 所用文件寄存器, 软元件初始值, 通过这个指令转换的程序的备注和当地软元件的使用都设置为“跟随 PLC 文件设置”。
- 然而, 如果“使用当地软元件”在 PLC 文件设定里指定, 程序通过这个指令转移, 如果程序内存的程序数量超过了参数设置程序的数量, 错误将会出现。
- 在使用通过这个指令转移的程序里的当地软元件时, 在参数里注册一个虚拟程序, 使用 PUNLOADP 指令来删除虚拟文件, 然后使用 PLOADP 指令来装载该程序。
- ② I/O 刷新设置
- 通过这个指令转移的程序的 I/O 刷新设置对于输入和输出都不可行。

- (10) “PLOADP” 指令和 “在运行时写入” 处理不能同时执行。
- (a) 在 PLOADP 指令的处理过程中, 需要一个运行时候的写入, 运行的写入会延迟。
在 PLOADP 指令处理结束后, 运行中的处理再开始。
- (b) 当 PLOADP 指令在运行中写入的时候执行, PLOADP 指令的处理将延迟。
在运行中处理结束后, PLOADP 指令的处理再开始。

[运行错误]

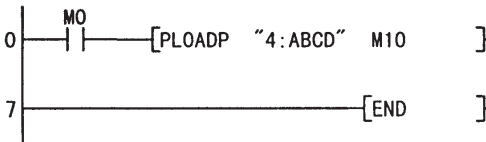
- (1) 在下列出现运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 文件名称在由⑤指令的驱动号码中不存在。(错误代码: 2410)
 - 由⑤指定的驱动号无效。(错误代码: 4100)
 - 在 0 驱动中, 没有足够的内存来装载指定的文件。(错误代码: 2413)
 - 下面显现的程序数字已经在程序内存中登记。(错误代码: 4101)
 - 在 SD720 中存储的程序号码已经使用, 或大于下面显示的最大程序数字。(错误代码: 4101)
 - 程序文件同已经存在的将装载的程序文件相同。(错误代码: 2410)
 - 本地驱动的文件大小不能保留。(错误代码: 2401)

CPU 类型名称	程序内存(文件号码)	最大程序号码
Q02 (H) CPU	28	28
Q06HCPU	60	60
Q12HCPU	124	124
Q25HCPU	124	124
Q12PHCPU	124	124
Q25PHCPU	124	124

[程序示例]

- (1) 当 M0 为 ON 时, 下列程序装载存储在驱动 4 中的文件 “ABCD. QPG” 到驱动 0, 同时程序处于备用状态。

[梯形图模式]



[列表模式]

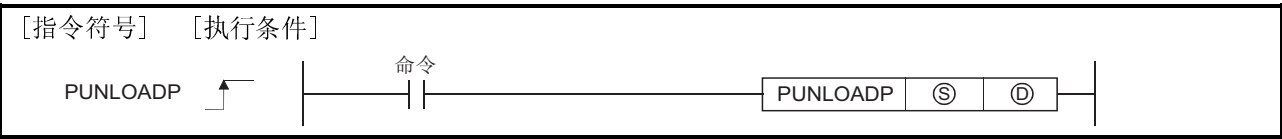
步	指令	软元件
0	LD	
1	PLOADP	M0
7	END	"4:ABCD" M10

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	×	×

9.6 从程序内存中卸载程序(PUNLOADP)

设定 数据	可用软元件								
	内部软元件		文件寄存器	MELSECNET/10 (H) 直接 J[]\K[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量	
	位	字		位	字			K, H	\$
⑤	—	○		—				—	○
⑥	△*	—		—				—	—

※: 局部软元件不能使用。

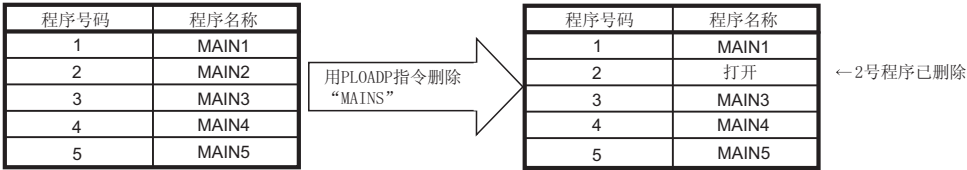


[设定数据]

设定数据	含义	数据类型
Ⓔ	将被卸载的程序文件名称的字符串数据或存储字符串数据的软元件的起始号码	BIN 16 位
Ⓕ	指令结束时软元件接通一个扫描周期	位

[功能]

- (1) 存储在程序内存(驱动 0)的备用程序将从程序内存中删除。
(用 PSCAN 指令设置为“扫描执行类型”的程序或用 PLOW 指令设置为“低速执行类型”的程序不能被删除。)
- (2) 被 PUNLOADP 指令删除的程序号码为 ON。
当程序号码为 1 到 5 的文件在 PLC 参数对话框的设定中已经设定,用 PUNLOADP 指令删除 2 号程序使得 2 号程序为 ON。



- (3) 对于文件名称不必指定扩展名(.QPG)
- (4) 由Ⓕ指令的位软元件在 PUNLOAD 指令结束的 END 程序处理中继续。在下一个 END 处理时,位软元件为 OFF。
- (5) PLOADP, PUNLOADP 和 PSWAPP 指令不能同时执行。
如果两个或更多的上述指令同时执行,随后需要执行的指令不能被执行。
当使用上述指令时,以用户来提供互锁避免同时执行。

- (6) 当 PLC 在为 OFF 后为 ON 或者 CPU 模块在 PUNLOADP 指令执行后复位设置, 下列操作将被执行。
- (a) 当在 PLC 参数对话框中已经进行了启动设置, 在启动设置位置的程序将转移到程序内存。
当用 PUNLOADP 指令删除的程序不能执行, 从启动设置和 PLC 参数对话框的程序设定中删除相应的文件名称。
 - (b) 当在 PLC 参数对话框的启动文件没有设定, “FILE SET ERROR” (错误代码:2400) 将出现。
 - ① 当用 PUNLOADP 指令删除的文件不能执行, 从 PLC 参数对话框的程序设定中删除相应程序名称。
 - ② 当用 PUNLOADP 指令删除的程序再次执行时, 把相应程序写入 CPU 模块。
- (7) 在中断程序里不要执行 PUNLOADP 指令。
(否则, 将导致功能故障)
- (8) 从程序内存中用这个指令将删除的程序将先用 PSTOP 指令设置为“待命执行类型”。
- (9) “PUNLOADP”指令和“在运行时写入”处理不能同时执行。
- (a) 在 PUNLOADP 指令的处理过程中, 需要一个运行时候的写入, 运行的写入会延迟。
在 PLOADP 指令处理结束后, 运行中的处理再开始。
 - (b) 当 PUNLOADP 指令在运行中写入的时候执行, PUNLOADP 指令的处理将延迟。
在运行中处理结束后 PUNLOADP 指令的处理再开始。

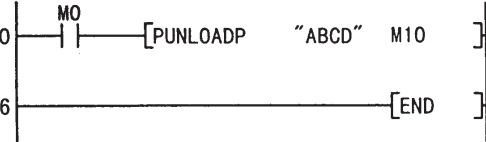
[运行错误]

- (1) 在下列出现运行错误的情况中, 错误标志(SM0)变为 ON, 并且错误代码存储在 SD0 中。
- 由⑤指令的文件名称不存在。 (错误代码: 2410)
 - 由⑤指定的程序不处于备用状态和正在被执行。 (错误代码: 4101)
 - 只有由⑤指定的程序在程序内存中存在。 (错误代码: 4101)

[程序示例]

(1) 当 M0 从 OFF 变为 ON 时, 下列文件从内存中删除存储在驱动 0 的“ABCD.QPG”文件。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	
1	PUNLOADP	M0
6	END	"ABCD" M10

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	×	×

9.7 装载+卸载(PSWAPP)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量	
	位	字		位	字			K, H	\$
Ⓔ1	—	○		—				—	○
Ⓔ2	—	○		—				—	○
Ⓓ	△*	—		—				—	—

※: 本地软元件不能使用。

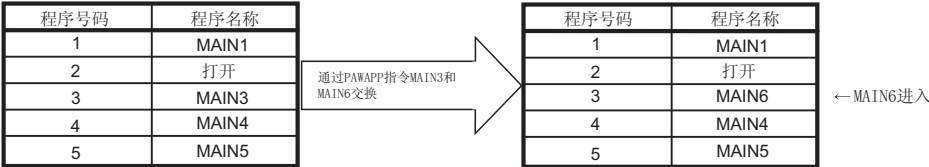


[设定数据]

设定数据	含义	数据类型
Ⓢ1	将被卸载的程序文件名称的字符串数据或存储字符串数据的软元件的起始号码	BIN 16 位
Ⓢ2	存储装载文件的驱动号码, 文件名称的字符串数据或存储字符串数据的软元件的起始号码	BIN 16 位
Ⓓ	指令的 ON 结束后, 软元件变为 ON 一个扫描周期	位

[功能]

- (1) 存储在程序内存(驱动 0)的备用程序将从程序内存中删除, 同时存储在内存卡或标准 ROM 中由 S2 指定的程序将转移到程序内存, 同时处于备用状态。
(当程序转移到程序内存, 程序必须有连续的空余空间。)
由 PACAN 指令设定为“扫描执行类型”的程序或者由 PLOW 指令设定为“低速执行类型”的程序不能删除。
- (2) 用 PSWAPP 指令转移到内存卡的程序和从程序内存中删除的程序的程序号码相同。
(如果在程序从程序内存删除之前有一个为 ON 的程序号码, 将要转移到程序内存的程序将没有为 ON 的程序号码。)
当程序号码 2 为 ON 时, 转移到程序内存的程序通过 PSWAPP 指令用程序交换方式注册为 3 号程序。



- (3) 驱动器 1, 2, 4 可以被指定 (驱动 3 不可以接受指令)
- 驱动 1: 内存卡 (RAM)
 - 驱动 2: 内存卡 (ROM)
 - 驱动 4: 标准 ROM
- (4) 对于一个文件名称不必指定一个扩展名 (.QPG) 。
- (5) 由 D 指定的位软元件在指令结束的 END 扫描处理过程中为 ON, 在下一个 END 处理的阶段为 OFF。
- (6) PLOADP, PUNLOADP 和 PSWAPP 指令不能同时执行。
如果两个或更多的上述指令同时执行, 随后需要执行的指令不能工作。
当使用上述指令时, 由用户来提供互锁避免同时执行。
- (7) 当 PLC 在为 OFF 后为 ON 或者 CPU 模块在 PUNLOADP 指令执行后复位设置, 下列操作将执行。
- (a) 当在 PLC 参数对话框中已经进行了启动设置, 启动设置位置的程序将转移到程序内存。
当用 PSWAPP 指令替换的程序执行时, 改变启动设置和 PLC 参数对话框中的程序设定为相应的程序名称。
- (b) 当在 PLC 参数对话框中没有进行启动设置, “FILE SET ERROR ” (错误代码: 2400) 将出现。
- ① 当被 PSWAPP 指令替换的程序被执行时, 改变 PLC 参数对话框的程序设定为相应的程序名称。
- ② 在 PLC 参数对话框的程序设置中执行程序设定时, 把相应程序重新写入 CPU 模块。
- (8) 在中断程序中不能执行这个指令。
(在中断程序中执行这个指令将导致故障)
- (9) 交换程序的 PC 文件设定如下:
- ① 每个程序的文件使用
所有文件寄存器, 软元件初始值, 注释和交换程序的本地软元件设定为 “跟随 PC 文件设定” 。
- ② I/O 刷新设置
对于交换程序的 I/O 刷新设置对于输入和输出都无法使用。
- (10) “PSWAPP” 指令和 “在 RUN 过程中写入” 处理不能同时执行。
- (a) 当在 PSWAPP 指令的处理过程中, 有一个 RUN 过程中的写入请求时, RUN 过程中的写入会延迟。
RUN 过程中的写入在 PSWAPP 指令处理结束后开始。
- (b) 当 PSWAPP 指令在 RUN 过程中写入的时候执行, PUNLOADP 指令的处理将延迟。
在 RUN 过程中的写入处理结束后, PSWAPP 指令的处理开始。

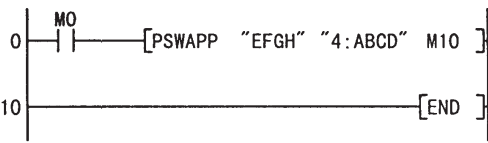
[运行错误]

- (1) 在下列出现运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 由⑤1和⑤2指定的驱动号码或文件名称不存在 (错误代码: 2410)
 - 由⑤1指定的驱动号码无效 (错误代码: 4100)
 - 在驱动 0 中没有足够的容量装载指定的程序 (错误代码: 2413)
 - 由⑤1指定的程序不处于备用状态或正在执行 (错误代码: 4101)

[程序示例]

- (1) 当 M0 由为 OFF 到为 ON, 下列程序将从内存中删除存储在驱动 0 中的文件 “EFGH.QPG”, 装载存储在驱动 4 中的文件 “ABCD.QPG” 到驱动 0, 同时置于备用状态。

[梯形图模式]



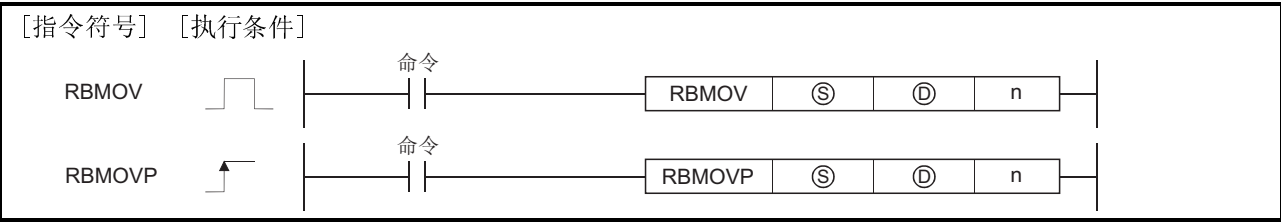
[列表模式]

步	指令	软元件
0	LD	M0
1	PSWAPP	"EFGH" "4:ABCD" M10
10	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
×	○	○	×	×

9.8 文件寄存器的高速块传送

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\K[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
⑤				○				—	—
⑥				○				—	—
n				○				○	—

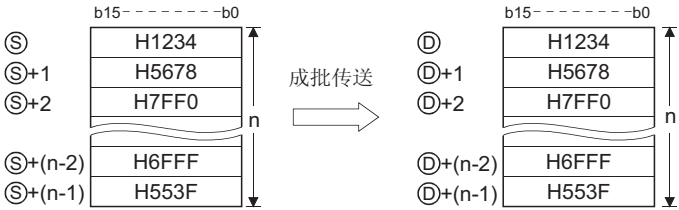


[设定数据]

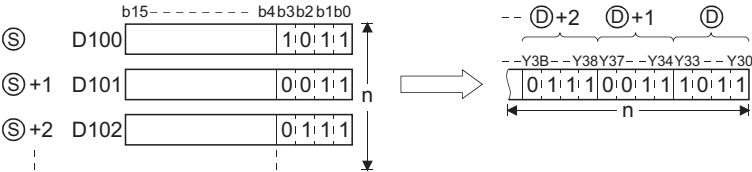
设定数据	含义	数据类型
⑤	存储传送数据软元件的起始号	BIN 16 位
⑥	存储目标软元件的起始号	
n	传送数据的数目	

[功能]

(1) 由⑤指定的软元件开始的“n”点 16 位数据成批传送到由⑥指定的软元件开始的“n”点区域。



- (2) 即使源和目标软元件之间有重叠，传送也可以。
对于传送到较小的软元件号，数据从⑤开始传送。对于传送到较大软元件号，数据从⑤+(n-1)开始传送。
- (3) 如果⑤是一个字软元件而⑥是一个位软元件，字软元件的对象是由位软元件数字定义指定的位数。
当 K1Y30 由⑥指定时，由⑤指定的字软元件较低四位作为对象。



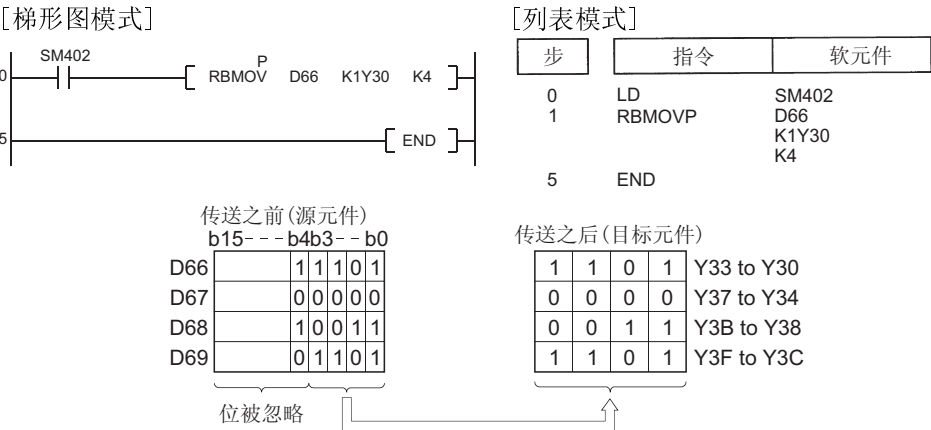
(4) 如果位软元件由⑤和⑥指定，⑤和⑥应一直有相同的数字位数。

[运行错误]

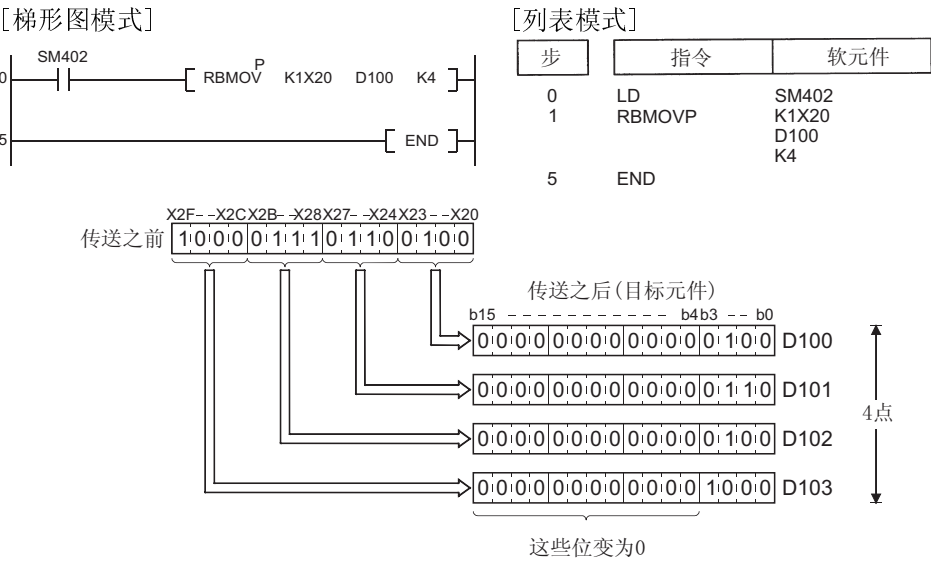
- (1) 在下列出现运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 始于⑤或⑥的“n”点软元件范围超过了现有软元件 (错误代码: 4101)
 - 文件寄存器不是由⑤或⑥指定的 (错误代码: 4101)

[程序示例]

(1) 下列程序以 4 个数据为一个单元的形式输出 D66~D69 的低 4 位数据到 Y30~Y3E。



(2) 下列程序以 4 个数据一个单元的形式从 X20~X2F 中输出数据到 D100~D103。



要点

RBMOV (P) 指令对于用 QnHCPU/QnPHCPU 成批传送大量文件寄存器数据很有用。对 QnCPU，该指令同 BMOV 相似。

RBMOW 和 BMOV 指令的处理速度对比表如下：

(1) 用 QnHCPU/QnPHCPU

对象内存 \ 指令	RBMOV		BMOV	
	100 字	1000 字	100 字	1000 字
SRAM	56.30μs	367.73μs	44.37μs	393.14μs
内置 RAM	44.37μs	393.14μs		
闪存卡	44.37μs	393.14μs		

(2) 用 QnCPU

对象内存 \ 指令	RBMOV		BMOV	
	100 字	1000 字	100 字	1000 字
SRAM	115.89μs	579.47μs	63.83μs	535.23μs
内置 RAM				
闪存卡				

9.9 写入自站 CPU 共享内存

ST0 或 T0 指令用来在多 CPU 系统中执行写入到自站 CPU 共享内存。ST0 或 T0 都可以用来执行写入 CPU 共享内存的操作。

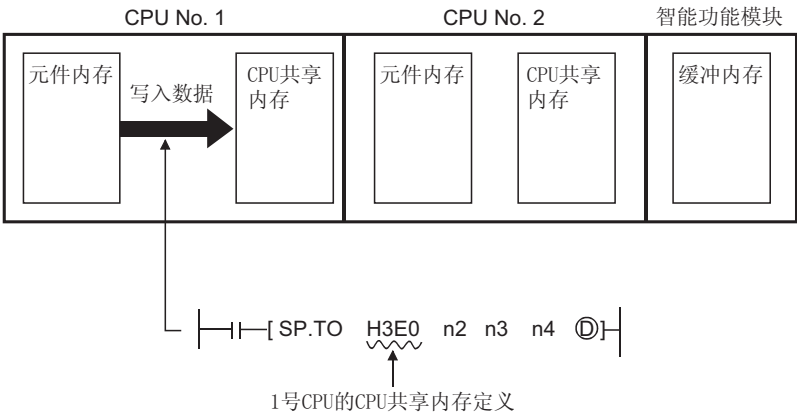
可以使用 T0 指令的 CPU 模块是 Q00CPU 和 Q01CPU。
其它任何 CPU 模块不能用 T0 指令执行写入到 CPU 共享内存的操作。
在 Q00CPU 或 Q01CPU 时执行写入到本地 CPU 共享内存时, 建议最好使用 T0 指令。

下列表格表明 ST0 和 T0 指令的可用性。

CPU 模块型号名称		ST0 指令	T0 指令
基本型 CPU	Q00CPU	可使用	可使用
	Q01CPU		
高性能型 QCPU	Q02 (H) CPU	可使用	不能使用
	Q06HCPU		
	Q12HCPU		
	Q25HCPU		
过程控制 CPU	Q12PHCPU	可使用	不能使用
	Q25PHCPU		

(1) ST0 指令的操作

ST0 指令能够写入数据到自站 CPU 模块的 CPU 共享内存。
下列数据表明当 ST0 指令在 1 号 CPU 执行时程序的处理过程。

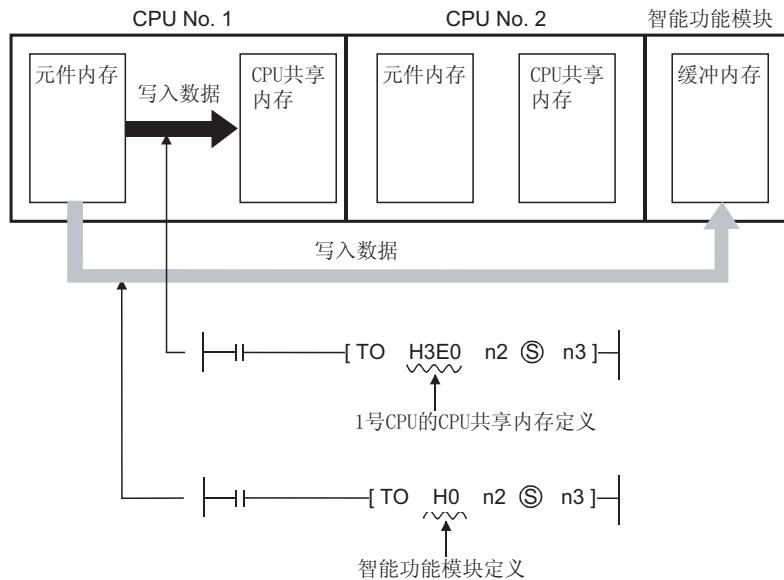


(2) T0 指令的操作

T0 指令能够写入软元件内存数据到下列内存。

- 自站 CPU 模块的 CPU 共享内存。
- 智能功能模块的缓冲内存。

下列数据表明当 T0 指令在 1 号 CPU 执行时的处理过程。



备注

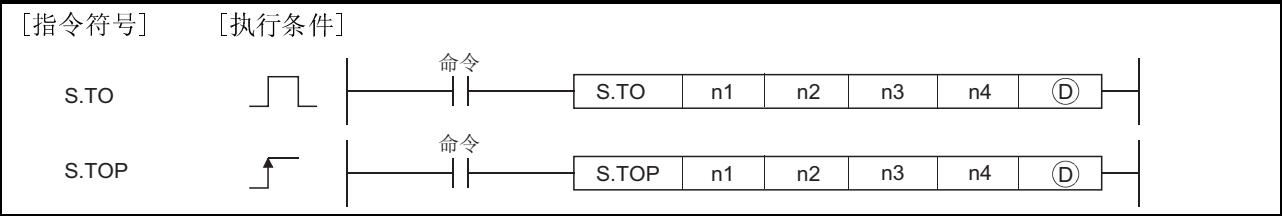
关于用 T0 指令写入到智能功能模块的缓冲内存，参考 7.8.2 节。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*1	△*2	○	×	×

*1：系列号前五位数为04101或之后的
Q00CPU/Q01CPU
*2：功能版本B或以后的版本

9.9.1 写入到自站 CPU 共享内存(S. TO (P))

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[] []		特殊功能模 块 U[] G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				U
n1	—	○		—				○	—
n2	—	○		—				○	—
n3	—	○		—				—	—
n4	—	○		—				○	—
⑤	○	○		—				—	—



[设定数据]

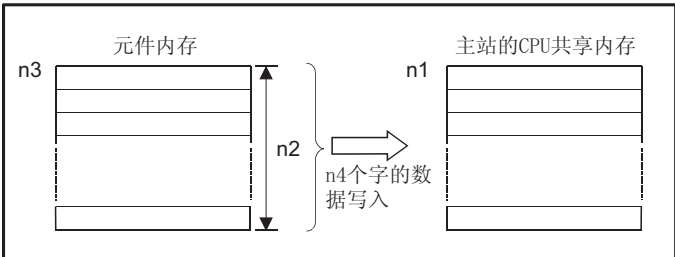
设定数据	含义	数据类型
n1	自站起始 I/O 号	BIN 16 位
n2	写入指定的自站的 CPU 共享内存地址 基本模型 QCPU：00H 到 1FFH 高性能型 QCPU，过程 CPU：800H 到 FFFH	
n3	存储写入数据软元件的起始号码	
n4	写入数据的数目 基本模型 QCPU：1 到 320 高性能型 QCPU，过程控制 CPU：1 到 256	
Ⓓ	在写入完成时自站软元件变为 ON 一个扫描周期	位

[功能]

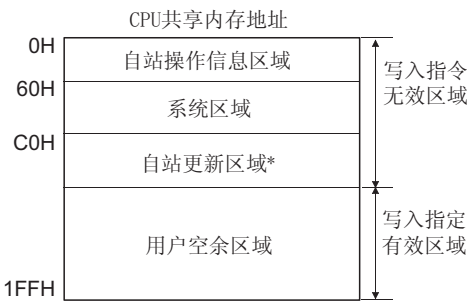
S. TO

- (1) 在软元件里写入 n4 个字的数据, 从自站 CPU 在 n3 中的号码开始到自站 CPU 模块在 n2 指定的共享内存地址。
当写入结束, 结束位变为 ON。

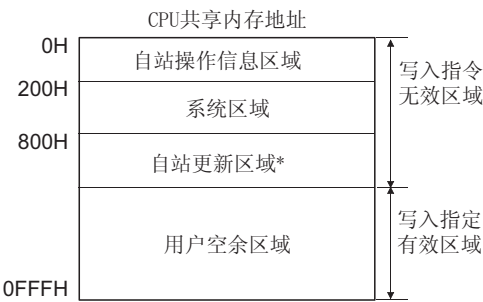
主机站



(a) 基本型 QCPU 的 CPU 共享内存地址



(b) 高性能型 QCPU 或过程控制 CPU 的 CPU 共享内存地址



※：当自动刷新设定没有使用时，作为用户空余区域使用。
当自动刷新设定完成时，自动开始传送范围和其后区域作为用户空余空间使用。

- (2) 当写入端数字是 0 时，处理没有执行同时结束软元件也不变为 ON。
- (3) 在每个站的一次扫描中只有一个 ST0 指令可以执行。
如果执行条件在两个或更多地方不能同时使用，随后执行的 ST0 指令由于自动握手而没有被处理。

要点	
(1) 从智能功能模块/特殊功能模块的缓冲内存中读取的数据可以通过使用智能功能模块/特殊功能模块软元件来执行。 对于智能功能模块/特殊功能模块软元件，参考高性能 QCPU (Q 模式) 的使用手册 (功能解释/程序基础) 或 QnACPU 编程手册 (基础篇)。	

[运行错误]

在下列出现运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。

- (1) 指定数据在下列范围之外 (错误代码: 4101)
- 写入点数 (n4) 不在 0 到 256 之间

● 写入目标自站的 CPU 共享内存地址 (n2) 的开始超出 CPU 共享内存地址范围

● 写入目标自站的 CPU 共享内存地址 (n2) + 写入点数 (n4), 超出了 CPU 的共享内存地址范围

● 存储写入数据的第一个软元件号 (n3) 加上写入点数 (n4), 超出软元件范围
- (2) 在 n1 里指定的数字不是自站的号码 (错误代码: 2107)
- (3) 在 n1 里指定的数字不包括 3E0H/3E10H/3E20H/3E30H (错误代码: 4100)
- (4) 指定的指令不合适 (错误代码: 4002)
- (5) 指定的软元件号码错误 (错误代码: 4003)
- (6) 不能使用的软元件被指定 (错误代码: 4004)

[程序示例]

- (1) 当 X0 为 ON 时, 存储 10 点数据的程序从在 D0 开始, 进入自站 CPU 共享内存的地址 800H。



备注

在 n1 里, 指定代表 CPU 模块插槽的起始 I/O 号的 4 个 16 进制数字的上面三位。

	CPU 插槽	插槽 0	插槽 1	插槽 2
起始 I/O 号	3E00	3E10	3E20	3E30
n1	3E0	3E1	3E2	3E3

QCPU			QnA	Q4AR
PLC CPU		处理CPU		
基本	高性能			
△*1	×	×	×	×

* 1 : 系列号前五位数为04101或之后的
Q00CPU/Q01CPU

9.9.2 写入到自站 CPU 共享内存(T0)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]\G[]		特殊功能模 块 U[]\G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				U
n1	—	○			○			○	○
n2	—	○			○			○	—
Ⓔ	—	○			—			○	—
n3	—	○			○			○	—

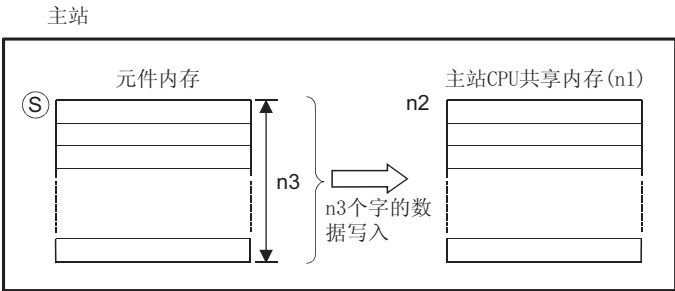
[指令符号]	[执行条件]
TO	
TOP	

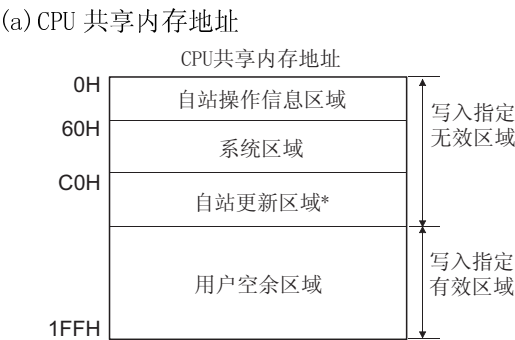
[设定数据]

设定数据	含义	数据类型
n1	自站起始 I/O 号	BIN 16 位
n2	写入目标自站的 CPU 共享内存地址	
Ⓢ	存储写入数据的软元件起始号	BIN 16 位
n3	写入数据的数目, 1 到 320	BIN 16 位

[功能]

- (1) 在软元件里写入 n3 个字的数据, 从自站 CPU 模块里Ⓢ中的数字开始, 到在 n2 中指定的 CPU 共享内存地址, 和其后的自站 CPU 模块区域。





※: 当自动刷新设定没有使用时, 作为用户空余区域使用。
当自动刷新设定完成时, 自动刷新传送范围和其后区域作为用户空余空间使用。

(2) 当写入数字是 0 时, 没有执行处理

要点	
智能功能模块软元件也可以用来执行 CPU 共享内存的数据写入。 对于智能功能模块软元件, 参考使用过的 CPU 模块的用户手册(功能解释, 程序基础)。	

[运行错误]

- 在下列出现运行错误的情况中, 错误标志(SM0)变为 ON, 并且错误代码存储在 SD0 中。
- (1) 指定数据在下列范围之外

(错误代码: 4101)

- 写入点数 (n3) 不在 0 到 320 之间
 - 写入目标自站的 CPU 共享内存地址 (n2) 的开始超过了 CPU 共享内存地址范围
 - 写入目标自站的 CPU 共享内存地址 + 写入点数 (n2), 超过了 CPU 的共享内存地址范围
 - 存储写入数据的起始软元件号 + 写入点数 (n3) 超过了软元件范围
- (2) 在 n1 里指定的 I/O 号不是自站的号码。

(错误代码: 2107)
- (3) 在 n1 里指定的数字不包括 3E0H。

(错误代码: 4100)
- (4) 指定的指令不合适。

(错误代码: 4002)
- (5) 指定的软元件号是错误的。

(错误代码: 4003)
- (6) 不能使用的软元件被指定。

(错误代码: 4004)

[程序示例]

(1) 当 X0 为 ON 时，程序从 CPU 共享内存的 C0 地址处存储 10 点的数据到本站。



备注

在 n1 里, 指定起始 I/O 号的前三个数字。用来代表 CPU 模块插槽的 4 位 16 进制数据。

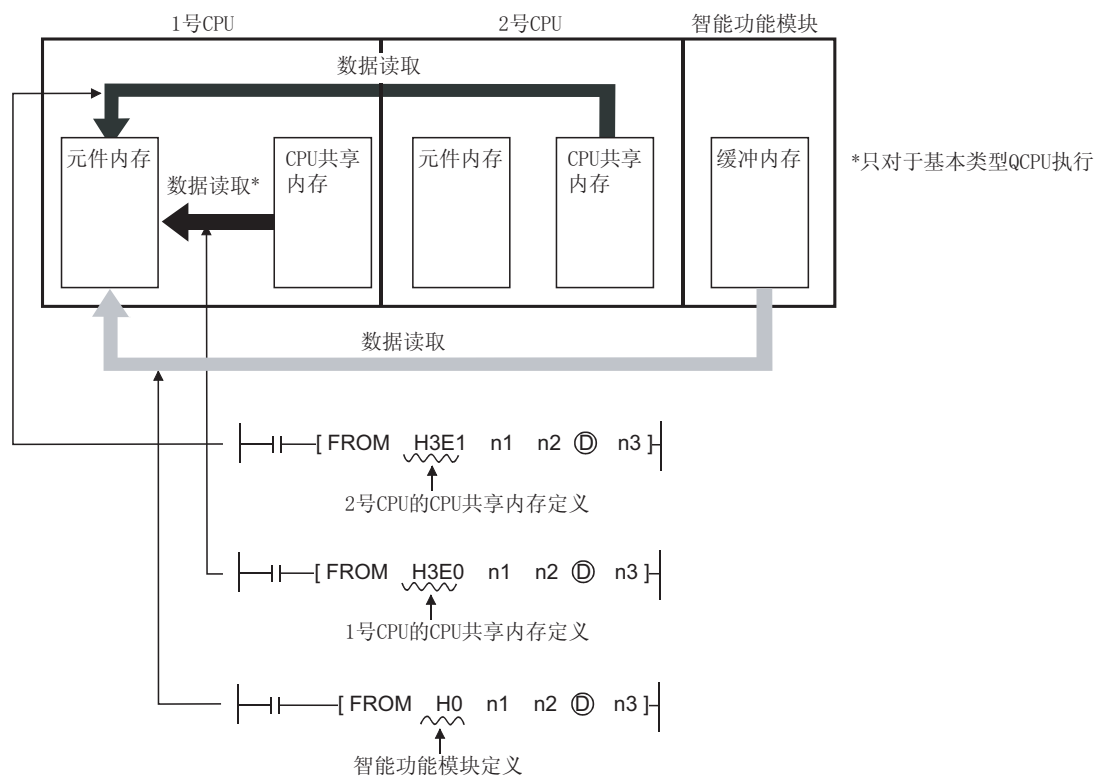
	CPU 插槽
起始 I/O 号	3E00
n1	3E0

9.10 从其他站的 CPU 共享内存中读取数据

多 CPU 系统的 FROM 指令可以从以下内存中执行读取

- 智能功能模块的缓冲内存
- 另一站 CPU 模块的 CPU 共享内存
- 本站 CPU 模块的 CPU 共享内存 (只能对于基本类型 QCPU 执行)

下列数据表明当 FROM 指令在 1 号 CPU 在执行时如何处理



备注

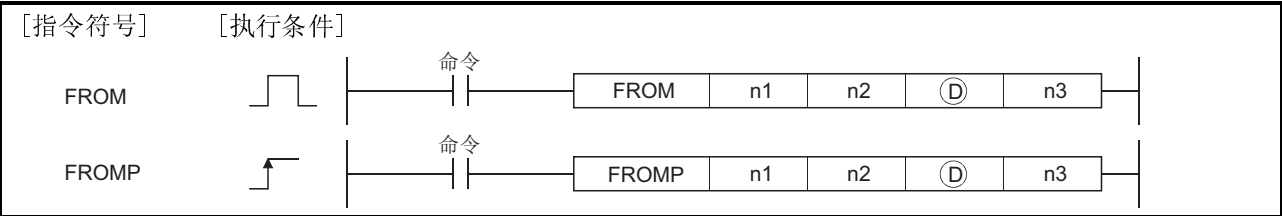
有关用 FROM 指令从智能功能模块的缓冲内存中读取数据，参考 7.8.1 节。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*1	△*2	○	×	×

* 1 : 系列号前五位数为04101或之后的
Q00CPU/Q01CPU
* 2 : 功能版本B或以后的版本

9. 10. 1 从其他站的共享内存读取数据 (FROM (P))

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[][]		特殊功能模 块 U[][G]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				U
n1	—	○			○			○	○
n2	—	○			○			○	—
⑤	—	○			—			—	—
n3	—	○			○			○	—

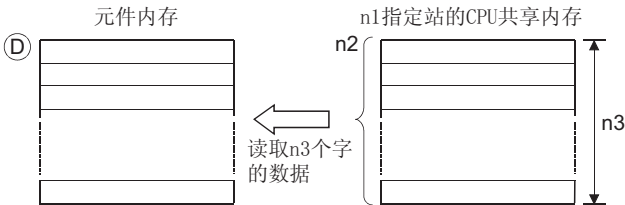


[设定数据]

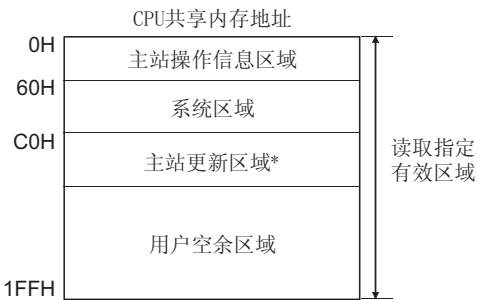
设定数据	含义	数据类型
n1	读取 CPU 模块的起始 I/O 号	BIN 16 位
n2	读取数据的起始地址	BIN 16/32 位
Ⓓ	存储读取数据的软元件起始号	BIN 16 位
n3	读取数据的数目 基本类型 QCPU : 1 到 512 高性能型 QCPU, 过程控制 CPU : 1 到 4096	

[功能]

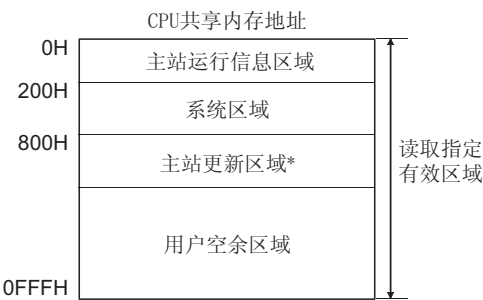
- FROM
- (1) 读取从由 n1 指定的 CPU 模块中, 从由 n2 指定的 CPU 共享地址中读取 n3 个字的数据, 同时存储在由Ⓓ指定的软元件其后的区域中。



(a) 基本型 QCPU 的 CPU 共享内存地址



(b) 高性能型 QCPU 和过程控制 CPU 的 CPU 共享内存地址



※: 当自动刷新设定没有使用时, 作为用户空余区域使用。
当自动刷新设定完成时, 自动刷新传送范围和其后区域作为用户空余空间使用。

(2) 当 0 指定在 n3 里作为读取数据的数目时, 不执行处理。

(3) 被读数据的数目根据目标 CPU 模块的不同而有所不同。
对于读取数据的数目, 可以参考使用过的 CPU 模式的用户手册 (功能解释, 编程基础。)

CPU 模式	读取点的数字
基本型 QCPU	1 到 512
高性能型 QCPU 过程控制 CPU	1 到 4096

要点	
(1) 从智能功能模块/特殊功能模块的缓冲内存中读取的数据可以通过使用智能功能模块/特殊功能模块软元件来执行。 对于智能功能模块/特殊功能模块软元件, 参考高性能 QCPU (Q 模式) 的使用手册 (功能解释/程序基础) 或 QNACPU 编程手册 (基本篇)。	

[运行错误]

- 在下列出现运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- (1) 指定数据在下列范围之外。

(错误代码: 4101)

● 执行读取的 CPU 共享内存地址 (n2) 的起始地址超出了 CPU 共享内存的范围。

● 执行读取的 CPU 共享内存地址 (n2) + 读取点数 (n3) 超出了 CPU 共享内存范围。

● 读取数据存储软元件数目 (D) + 读取点数 (n3) 超出了指定的软元件范围。
- (2) 智能功能模块指定的起始 I/O 号是自站的号码。

(错误代码: 2114)
- (3) 智能功能模块和 CPU 模块在指定使用智能功能模块的输入输出。

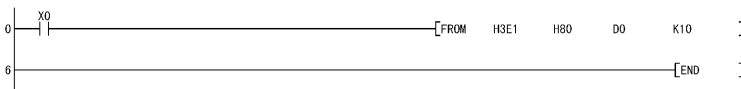
起始号的位置不存在。

(错误代码: 2110)

[程序示例]

- (1) 当 X0 为 ON 时, 下面的程序将 2 号站的 CPU 共享内存中从地址为 800H 开始的 10 个点数据存储到 D0 和其后的区域内。

[梯形图模式]



[列表模式]

```
0 LD      X0
1 FROM    H3E1  H80  D0  K10
6 END
```

备注

在 n1 里, 指定代表 CPU 模块的插槽的起始 I/O 号的 4 个 16 进制数字的上面三位。

	CPU 插槽	插槽 0	插槽 1	插槽 2
起始 I/O 号	3E00	3E10	3E20	3E30
n1	3E0	3E1	3E2	3E3

对于从 FROM 到 TO 指令, QCPU 提供了自动互锁。

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本型	高性能型			
△*1	△*2	○	×	×

*1 :系列号为04122或之后的
*2 :系列号为04012或之后的

9.11 选择性刷新指令 (COM)

下列 CPU 模块的 COM 指令可以参考 7.6.8 节。

- 系列号 04121 或更早的基本类型 QCPU
- 系列号 04011 或更早的高性能类型 QCPU
- CPU 处理

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 (H) 直接 J[]X[]		特殊功能模 块 U[]G[]	变址寄存器 Zn	常量 K, H	其它
	位	字		位	字				
—	—								



[功能]

- (1) 当 COM 指令执行时，将执行下列刷新操作
- I/O 刷新
 - CC-Link 的刷新
 - MELSECNET/H 的刷新
 - 智能功能模块的自动刷新
 - CPU 共享内存的自动刷新
 - 一般数据处理

备注

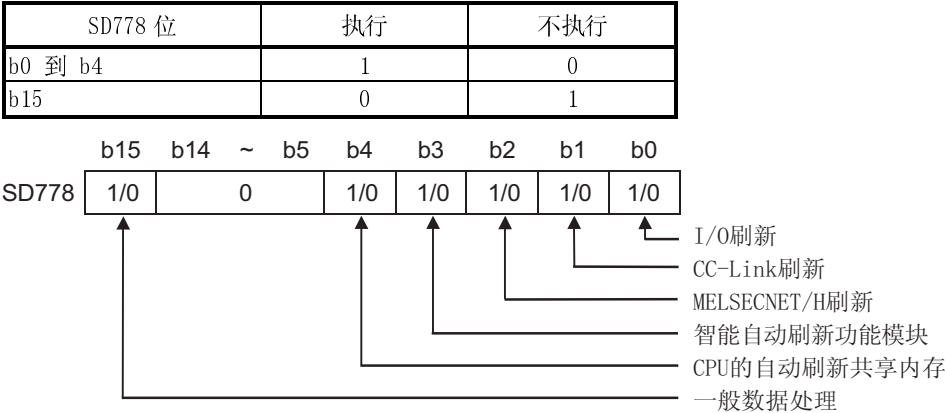
- 在一般数据处理中，将执行下列操作。
- 与外围软元件的沟通
 - 其它站监视
 - 通过系列交流模式从另一智能功能模块读取缓冲内存

- (2) 将 SM775 变为 OFF，以刷新除了 I/O 刷新之外的所有刷新条目。

- (3) 当选择刷新条目时
- (a) 通过 SD788 选择刷新条目，同时设置 SM775 到 ON
- 下图说明了可以通过将 SM775 变为 ON 和使用 ON/OFF SD778 来指定的刷新条目。

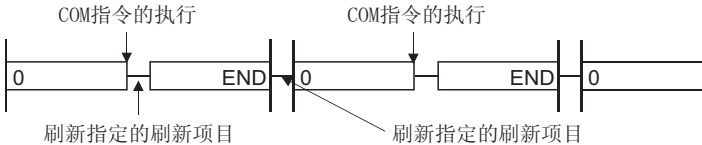
刷新条目	当 SM775 为 OFF	当 SM775 为 ON
I/O 刷新	执行	执行或未执行可以选择
CC-Link 刷新		
MESLECNET/H 刷新		
智能功能模块的自动刷新		
CPU 共享内存的自动刷新		
一般数据处理		

- (b) 使用 SD778 的 b0 到 b4 和 b15 来选择刷新条目。
每个 SD778 位的执行或非执行都能按如下指示：



例如：为了只使远程 I/O 站的发送/接收处理更快，只用 MELSECNET/H 刷新。（只设定 SD778 的 b2 和 b15 到 1(SD778：H8004)。)

- (4) 在 COM 指令执行的时候,CPU 模块中止了顺序程序的处理,同时刷新指定的刷新条目。
然而,它不执行 MELSECNET/H 和 MELSECNET/10 的低速循环刷新。



- (5) 在顺序程序里 COM 指令可以使用任意次数。
然而，需要注意使用 COM 指令由于一般数据处理或智能功能模块的自动刷新(包括链接刷新)时间，而延长了顺序程序的扫描时间。
- (6) 下列表列出了对于系列号“04012”或更高的高性能类型 QCPU 的 COM 指令的刷新条目

CPU 模块 类型名称	功能版本	系列号	SM775	刷新条目
Q00JCPU Q00CPU Q01CPU	A	04021 或更早	OFF	刷新所有的刷新条目
			ON	只刷新一般数据处理
	B	04122 或更迟	OFF	刷新所有的刷新条目
			ON	刷新由 SD778 选择的刷新条目
Q02CPU	A	—	ON/OFF	刷新所有的刷新条目
Q02HCPU Q06HCPU Q12HCPU Q25HCPU	B	04011 或更早	OFF	刷新所有的刷新条目
			ON	只刷新一般数据处理
		04012 或更迟	OFF	刷新所有的刷新条目
			ON	刷新由 SD778 选择的刷新条目

要点

在低速执行类型程序, 循环执行类型程序或中断程序里, COM 指令不能使用

10. 冗余系统指令(对于 Q4ARCPU)

下列是现有的冗余的系统指令(对于 Q4ARCPU)

分类	指令	含义	参考章节
CPU 启动时操作模式设定指令	S. STMODE	<ul style="list-style-type: none"> 当电源用 Q4ARCPU 启动打开时, 无论是在打开之前清除 Q4ARCPU 或不清除, 需要由⑤指定操作模式 	10.1 节
CPU 转换时间操作模式设定指令	S. CGMODE	<ul style="list-style-type: none"> 当控制器从控制系统转换到备用系统时, 无论是在打开之前清除 Q4ARCPU 或不清除, 需要⑤指定操作模式 	10.2 节
数据追踪指令	S. TRUCK	<ul style="list-style-type: none"> 在 END 指令的处理过程中, 与存储在从由⑤指定的软元件开始的区域的参数块数据内容保持一致, 执行软元件存储追踪 	10.3 节
缓冲存储批刷新指令	S. SPREF	<ul style="list-style-type: none"> 与存储在从由⑤指定的软元件开始的区域的参数块数据内容保持一致, 批读取/写入特殊功能模块缓冲的内容 	10.4 节

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本性	高性能性			
×	×	×	×	○

10.1 CPU 启动过程中的操作模式设定指令(S. STMODE)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[][]		特殊功能模 块 U[][G]	变址寄存器 Zn	常量	其它
	位	字		位	字				
①	—						○	—	
②	—						○	—	



[设定数据]

设定数据	含义	数据类型
①	操作模式设定	BIN 16 位
②	初始启动切换电源超时	

[功能]

- (1) 当电源打开并且 CPU 模块启动时,可以指定 CPU 软元件是否清除和启动或者不清除和启动。这个指令可以通过①来执行。
- (2) 当指定了热启动模式,在某一特定时间出现了电源暂时中断,可以执行自动数据清除和重新启动。在这种情况下,在②下指定转换电源超出时间。
- (3) 当电源打开时执行这个指令。
由于这个原因,即使指令触点关闭也没有问题。这个指令触点变成虚触点。当指令触点在程序执行时打开,NOP 处理将执行。
- (4) 在每个系统里只能建立其中一个指令。如果有多重程序文件,这个指令只能建立在一个文件里。如果其中的指令多于一个,操作不能保证。
- (5) 在①里通过设定 0 或 1 可以执行
0: 初始启动模式(清除锁存范围外的软元件)
1: 热启动模式(不清除*1)
*1: 清除下列信息
 - 变址寄存器
 - 信号流(操作结果)此外,预设特殊继电器 SM 和特殊寄存器 SD。

- (6) 在 S2 里可以从 0 到 65535 进行指定(单位 S)
当指定为 0 时, 初始启动模式转换不能执行。如果一个超过 32767 的数字被设定, 在执行时使用 16 进制数。

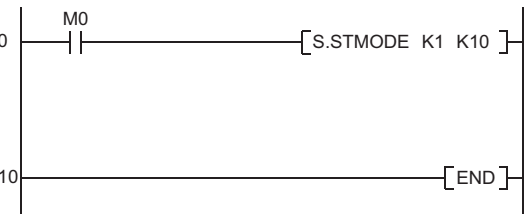
[运行错误]

- (1) 在下列出现运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SDO 中
- 当超过指定允许范围的值被指定 (错误代码: 4104)

[程序示例]

- (1) 当电源打开同时设定暂时转换电源超出时间为 10s 时, 程序在热启动模式下启动 CPU 模块。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	M0
1	S.STMODE	K1 K10
10	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本性	高性能性			
×	×	×	×	○

10.2 CPU 切换时间操作模式设定指令(S. CGMODE)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[]C[]		特殊功能模 块 U[]G[]	变址寄存器 Zn	常量	其它
	位	字		位	字				
⑤	—						○	—	

[指令符号]	[执行条件]
S.CGMODE	<div> 命令 <div>S.CGMODE</div> <div>⑤</div> </div>

[设定数据]

设定数据	含义	数据类型
⑤	操作模式设定	BIN 16 位

[功能]

- (1) 当控制器从控制系统切换到备用系统时, 指定 CPU 软元件是否被清除和切换或者不清除和切换。这个指定在⑤下执行。
- (2) 当电源打开时这个指令从 STOP 转变到 RUN 执行。因为这个原因, 即使这个指令触点关闭也没有问题。这个指令触点将变成虚触点。当指令触点在程序执行时打开, NOP 处理将执行。
- (3) 在一个系统里只能建立其中一个指令。即使有多重程序文件, 只能在一个文件里建立这个指令。如果其中不只一个指令存在, 操作不能保证。
- (4) 在⑤里设定 0 或 1
 - 0: 初始启动模式(清除闭锁范围外的软元件)
 - 1: 热启动模式(不清除 *1)
 - *1:
 - 全部信号流(操作结果)打开
 - 预设特殊继电器 SM 和特殊寄存器 SD

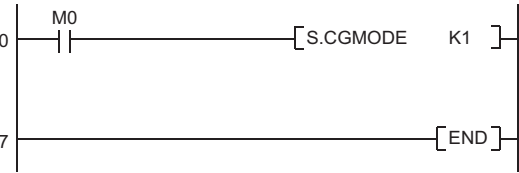
[运行错误]

- (1) 在下列出现运行错误的情况中, 错误标志(SM0)变为 ON, 并且错误代码存储在 SD0 中。
 - 当一个超过指定范围的值被指定 (错误代码: 4104)

[程序示例]

(1) 当从控制系统切换到备用系统，这个程序在热启动模式下启动 CPU 模块。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	M0
1	S.CGMODE	K1
7	END	

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本性	高性能性			
×	×	×	×	○

10.3 数据追踪指令(S. TRUCK)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J [][]		特殊功能模 块 U [][] G [][]	变址寄存器 Zn	常量	其它
	位	字		位	字				
⑤	—	○*	○	—					

*: 局部软元件不能使用

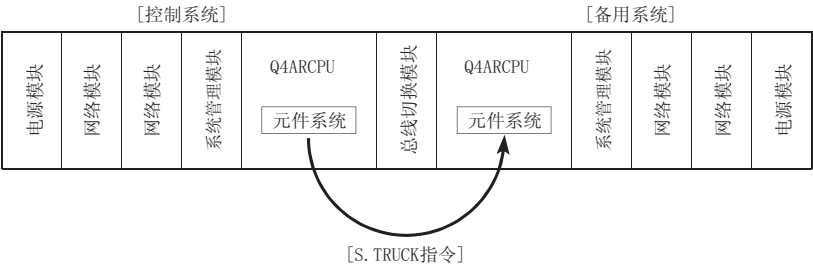


[设定数据]

设定数据	含义	数据类型
⑤	参数块起始软元件	BIN 16 位

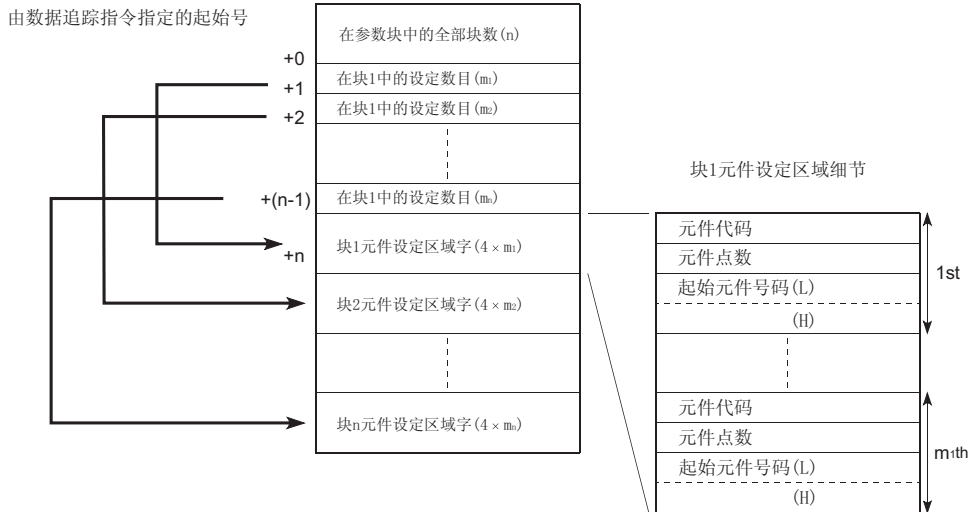
[功能]

- 对于每个通过这个指令执行的扫描，在 END 处理的过程中，Q4ARCPU 跟随由⑤指定的软元件中存储的参数块数据内容，执行软元件数据存储追踪。
* 追踪是一个从控制系统 Q4ARCPU 软元件存储转移到备用系统 Q4ARCPU 软元件系统的功能。运输在 END 处理的过程中执行。



- 当电源打开同时系统从复位中启动，Q4ARCPU 读取参数块内容。(因此，如果参数块内容改变，系统必需重新启动。) 参数块由多个块构成。在指定要执行的块特殊继电器从 SM1520 到 SM1583 执行和特殊继电器关闭后，执行这个指令。
- 在一个系统里只能建立其中一个指令
即使有多个程序文件，只能在一个文件里建立这个指令。如果多于一个文件存在，操作不能保证。

(4) 参数块有如下的构成



该块设定了软元件存储类型, 触点, 起始号码等。块追踪对应于传送触发器 SM1520 到 1583 的为 ON 的传送触发器块的内容 (SM1520→块 1, SM1521→块 2, ..., SM1583→块 64)。

要点
数据追踪指令传送触发器 SM1520 到 SM1583 可以使用同样的传送触发器号码作为缓冲存储器批更新指令。

参数块设定指令内容

(a) 参数块的全部块数

参数块是一个多个块的集合。设定有多少块可以使用。

(b) 设置块 n 的数目

对于每个块的软元件设定区域, 设定了设定软元件的数目。

(c) 块 n 软元件设置区域

这个可以设定实际追踪数据的软元件存储器的类型, 点数, 起始软元件号码。

① 软元件代码设定软元件存储器的类型

设定如下:

软元件	软元件代码	软元件	软元件代码	软元件	软元件代码	软元件	软元件代码
X	0	B	5	C	10	Z	15
Y	1	F	6	D	11	SB	16
M	2	V	7	W	12	SW	17
L	3	ST	8	R	13	SM	18
S	4	T	9	ZR	14	SD	19

T/C 包括触点号码, 线圈和当前值。

- ② 触点的软元件号码设定了将要追踪的软元件号码。
设定可以在十进制或十六进制下执行。
位软元件设定为 16 的倍数。
- ③ 起始软元件号码设定软元件存储器的起始号码。
设定可以在十进制或十六进制下执行。
位软元件设定为 16 的倍数 (0, 16, 32, ...)

备注

当设定参数块时,应用下列限制。

- $m_1 + m_2 + \dots + m_n \leq 2048$ 。 $n \leq 64$ 。
- 当设定区域从 m_1 到 m_n 是 0 时, 块的数目设定为 $m=0$ 。
对于设定不能执行的块数字设定为 0, 使得有可能跳越一个块。
- 在一个扫描 END 得处理过程中, 对于执行追踪的在一个块的点数: 块全部追踪点数字 $\leq 48k$ 字。如果超过 48K 字, 错误将被探测到同时追踪不能执行。
- 当指定一个位软元件作为执行追踪的软元件, 设定软元件的点数和起始软元件号码为 16 的倍数。
- 当计时器和计算器作为追踪的软元件被指定, 下面的公式用来计算追踪软元件的实际数目。
追踪软元件的点数 = 设定软元件的点数 $\times (1 + 1/8)$ 。

(5) 追踪执行模式的两种类型如下。可以使用继电器 SM1518 来选择。

在扫描 END 处理让 SM1518ON/OFF 后, 选择内容有效。

(a) 批通讯模式 (SM1518: 为 OFF 时)

如果当追踪执行时, 备用系统使用追踪存储器时, 控制系统在等待备用系统处理结束后执行追踪处理。如果控制系统 CPU 产生追踪处理等待时间, 这个时间累积将增加扫描时间。

(b) 在重复模式里 (SM1518: 为 ON 时)

当追踪执行时, 备用系统使用追踪存储, 控制系统在没有执行追踪处理的情况下, 重复执行 END 处理。当追踪处理被重复时, 下列追踪要求不能接收. 控制系统 CPU 不能产生追踪处理等待时间, 所以扫描时间不会延长。

(6) 当一个特定的块追踪处理已经结束, Q4ARCPU 的追踪结束标志变为 ON1 个扫描周期。

追踪结束标志对应每个块和从 SM1712 到 SM1775 的特殊继电器。(块 1 \rightarrow SM1712, 块 2 \rightarrow SM1713, ..., 块 64 \rightarrow SM1775)

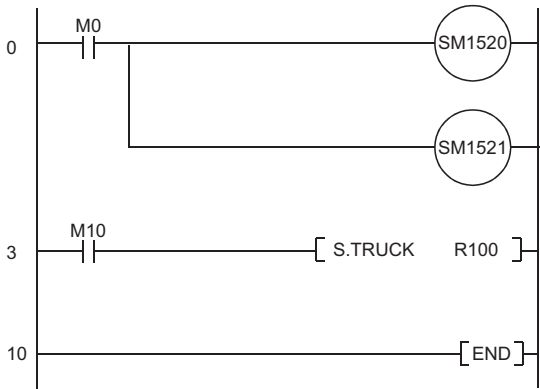
[运行错误]

- (1) 在下列发生运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 即使当文件寄存器 R 在指定的参数块被指定, 文件寄存器的文件也不存在 (错误代码: 2402)
 - 当超过指定范围的值被指定 (错误代码: 4104)
 - 当追踪软元件的点数超过 48K 个字 (错误代码: 4104)

[程序示例]

(1) 该程序从存储在 R100 的参数块 1 和 2 执行追踪。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	M0
1	OUT	M1520
2	OUT	M1521
3	LD	M10
4	S.TRUCK	R100
10	END	

(2) 这是个当内部继电器 M0 到 M95, M320 到 M639 和数据寄存器 D0 到 D29, D600 到 D699 被追踪的示例参数块。

R100	K2	参数块总块数	
R101	K2	块1的设定数目	
R102	K2	块2的设定数目	
R103	K2	元件代码(M)	
R104	K96	元件点数(96点)	
R105	K0	起始元件号码(M0)	块1元件设定区域
R106	K0		
R107	K2	元件代码(M)	
R108	K320	元件点数(320点)	
R109	K320	起始元件号码(M320)	块2元件设定区域
R110	K0		
R111	K11	元件号码(D)	
R112	K30	元件点数(30点)	
R113	K0	起始元件号码(D0)	块2元件设定区域
R114	K0		
R115	K11	元件代码(D)	
R116	K100	元件点数(100点)	
R117	K600	起始元件号码(D600)	
R118	K0		

- (a) 块 1 软元件设定区域
指定内部继电器 M0 到 M95, M320 到 M639
- (b) 块 2 软元件设定区域
指定数据寄存器 D0 到 D29, D600 到 D699

QCPU			QnA	Q4AR
PLC CPU		过程控制CPU		
基本性	高性能性			
×	×	×	×	○

10.4 缓冲内存批刷新指令(S. SPREF)

设定 数据	可用软元件								
	内部软元件 (系统, 用户)		文件寄存器	MELSECNET/10 直接 J[][]		特殊功能模 块 U[][]G[][]	变址寄存器 Zn	常量	其它
	位	字		位	字				
⑤	—	○*	○	—					

*: 只用于闭锁软元件

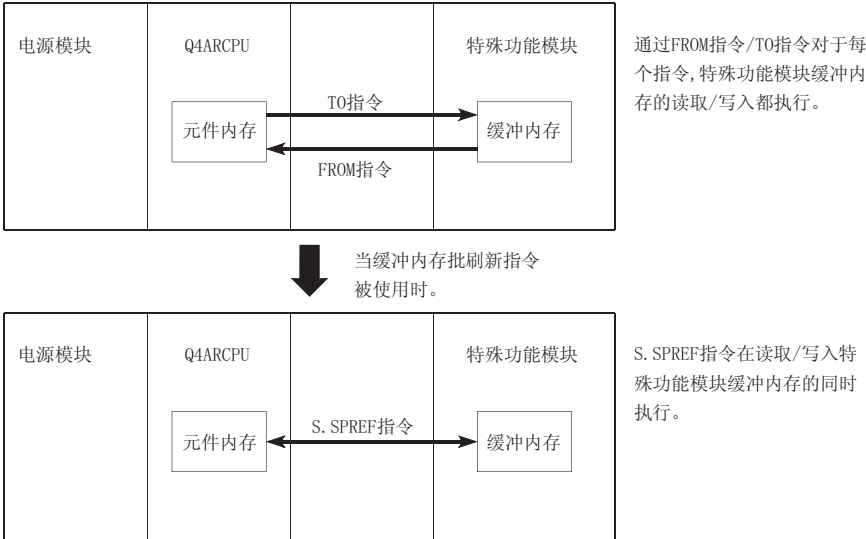


[设定数据]

设定数据	含义	数据类型
⑤	参数块起始软元件	BIN 16 位

[功能]

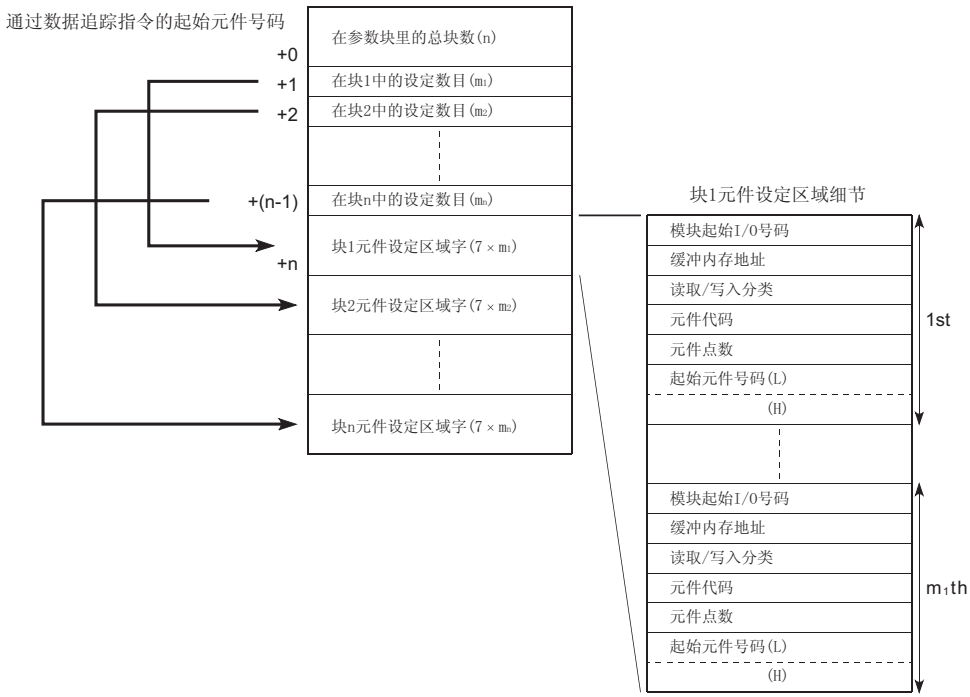
- (1) 特殊功能模块缓冲内存的内容与存储在⑤指定的软元件和其后区域里的参数块数据的内容相一致, 都是批读取/写入。



要点
<ul style="list-style-type: none">对于在 MELSECNET (II), /B, /10 中的远程 I/O 站的特殊的功能模块, 缓冲内存批刷新指令不能执行。对于在 MELSECNET/MINI-S3 中的特殊功能模块, 缓冲内存批刷新指令不能执行。

(2) 在该指令执行前设定参数块内容。参数块由多个块组成。该指令在将要执行的块的特殊继电器 SM1520 到 SM1583 设定后执行。(特殊继电器变为 ON)。

(3) 参数块构成如下：



该块用来设定执行批刷新, 特殊功能模块缓冲内存和缓冲内存的特殊功能模块的读/写软元件内存类型, 点数, 起始号码等。
根据传送触点 SM1520 到 SM1583, 该块根据变为 ON 的传送触点块的内容执行批刷新。(SM1520→块 1, SM1521→块 2, ..., SM1583→块 64)。

要点	
缓冲内存批刷新指令传送触点 SM1520 到 SM1583 能够同数据传送指令一样, 使用同样的传送触点号码。	

参数块设定项目内容

- (a) 参数块总块数
参数块是一个多个块的集合。这可以设定有多少块被使用。
- (b) 块 n 的设定数目
这可以设定每个块软元件的设定区域的软元件的数目。

(c) 块 n 的软元件设定区域

这对于将被批刷新的特殊功能模块的缓冲内存和软元件内存，可以用来实际设定其类型，点数和起始软元件号码。

① 模块起始 I/O 号码设定了目标特殊功能模块的起始 I/O 号码。
当数字用 3 位的 16 进制数表示时，设定时取其前两位数。（当 X/Y100 时，设定到 H10）

② 缓冲内存地址设定了需要刷新的特殊功能模块的缓冲内存起始地址。

③ 读取/写入分类设定是否读取或写入缓冲内存。
该设定是：0:读取，1: 写入

④ 软元件代码设定了软元件内存类型
该设定如下表：

软元件	软元件代码	软元件	软元件代码	软元件	软元件代码	软元件	软元件代码
X	0	B	5	C	10	Z	15
Y	1	F	6	D	11	SB	16
M	2	—	—	W	12	SW	17
L	3	ST	8	R	13	SM	18
—	—	T	9	ZR	14	SD	19

T/C 只是当前值。

⑤ 软元件数设定了追踪的软元件数目。
该设定用十进制或十六进制。
位软元件设定为 16 的倍数。

⑥ 起始软元件号码设定了软元件内存起始号码。
该设定用十进制或十六进制。
位软元件用 16 的倍数设定 (0, 16, 32...)

备注

当设定参数块时, 应用如下限制

- $m_1 + m_2 + \dots + m_n \leq 256$ 。 $n \leq 64$ 。
- 当 m_1 到 m_n 设定区间是 0 时, 块数字设定为 $m = 0$
当没有块设定时, 块数设定为 0, 跳越这些块。

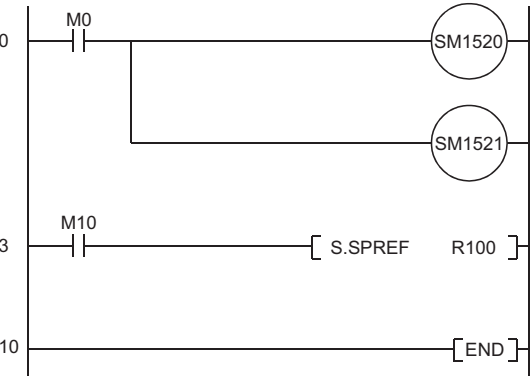
[运行错误]

- (1) 在下列出现运行错误的情况中, 错误标志 (SM0) 变为 ON, 并且错误代码存储在 SD0 中。
- 当超过指定范围的值被指定 (错误代码: 4104)

[程序示例]

(1) 该程序从存储在 R100 和其后的紧随参数块的块 1 和块 2 刷新缓冲内存。

[梯形图模式]



[列表模式]

步	指令	软元件
0	LD	M0
1	OUT	M1520
2	OUT	M1521
3	LD	M10
4	S.SPREF	R100
10	END	

(2) 这是一个参数块设定在 X/Y20 特殊功能模块缓冲内存从 0 到 3, 10 到 11 的内容和 X/Y100 特殊功能模块缓冲内存 110 到 119 内容被刷新时的例子。

R100	K2	参数块总块数	
R101	K2	块1设定数目	
R102	K1	块2设定数目	
R103	H2	模块起始I/O号码(X/Y20)	
R104	K0	缓冲内存地址(地址0)	
R105	K0	读取/写入类别(读取)	
R106	K13	元件代码(R)	
R107	K4	元件点数(4点)	
R108	K0	起始元件号码(R0)	
R109	K0		
R110	H2	模块起始I/O号码(X/Y20)	
R111	K10	缓冲内存地址(地址10)	
R112	K1	读取/写入类别(写入)	
R113	K13	元件代码(R)	
R114	K2	元件点数(2点)	
R115	K10	起始元件号码(R10)	
R116	K0		
R117	H10	模块起始I/O号码(X/Y100)	
R118	K110	缓冲内存地址(地址110)	
R119	K0	读取/写入类别(读取)	
R120	K11	元件代码(D)	
R121	K10	元件点数(10点)	
R122	K110	起始元件号码(D110)	
R123	K0		

块1元件设定区域

块2元件设定区域

① 块 1 软元件设定区域

- 读取 X/Y20 特殊功能模块缓冲内存 0 到 3 内容到 R0 到 R3。
- 写入 R10 和 11 的内容到 X/Y20 的特殊功能模块缓冲内存 10, 11。

② 块 2 软元件设定区域

- 读取 X/Y100 特殊功能模块缓冲内存 110 到 119 内容到 D110 到 D119

备忘录

[illegible]

11 错误代码

11

如果在 PLC 电源打开时出现错误,当 CPU 模块在 RUN 状态或当其运行时, QCPU/QNACPU 通过使用自诊断功能显现错误(LED 显示,在显示器上有信息),随后在特殊继电器 SM, 和特殊寄存器 SD 中存储错误信息。

以下信息包括可能出现在 QCPU/QNACPU 上的故障,以及与之对应的修正措施。

注释

有关编程工具,智能功能模块/特殊功能模块或网络系统提出通用数据处理请求时发生的故障,可以参考以下手册查询错误代码(4000H 至 4FFFH)。

Q 系列:使用的 CPU 模块的用户手册(硬件设计,维修和检查)。

QnA 系列:使用 CPU 模块的用户手册(详细篇)。

11.1 如何读取错误代码

当错误出现时,在能够执行 GPP 功能的外围设备中,可以读取错误代码,错误信息和其它信息
参考 GPP 功能外围设备的操作手册,了解操作外围设备的具体细节。

11.2 错误代码列表

11.2.1 基本型 QCPU 错误代码列表

下列信息是关于错误代码和错误信息意义、原因和修改措施的。

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15) * 1	个别信息 (SD16 到 26) * 1	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
1000	MAIN CPU DOWN	—	—	关闭	闪烁/打开	停止	一直	
1001								
1002								
1003								
1004								
1005								
1006								
1009								
1010	END NOT EXECUTE	—	—	关闭	闪烁	停止	当执行 END 指令。	
1011								
1012								
1020	SFCP. END ERROR	—	—	关闭	闪烁	停止	当 SFC 程序执行时	
1101	RAM ERROR	—	—	关闭	闪烁	停止	在电源 ON/复位时	
1102								
1103								
1104								
1105								
1200	OPE. CIRCUIT ERR.	—	—	关闭	闪烁	停止	在电源 ON/复位时	
1201								
1202								
1300	FUSE BREAK OFF	单元号码	—	关闭/打开	闪烁/打开	停止/继续 *2	当执行 END 指令时	
1310	I/O INT ERROR	—	—	关闭	闪烁	停止	处于中断时中	
1311								
1401	SP. UNIT DOWN	单元号码	—	关闭	闪烁	停止/继续 *3	在电源 ON/复位 当智能功能模块被访问时	
1402			程序错误位置				当智能功能模块访问指令被执行时。	
1403			—				当 END 指令执行时。	

*1 在圆括号里的字符表示存储个别信息的特殊寄存器的号码。

*2 在错误出现时，CPU 的操作状态能在参数上设定。(LED 显示会相应地改变)

*3 通过设定参数，每个模块的关闭/继续操作是可选的。

错误代码 (SD0) *1	错误内容和原因	修改方法	相应 CPU
1000	主 CPU 的运行模式的挂起或故障。 (1) 由于噪音和其它原因引起的故障。 (2) 硬件故障。	<ul style="list-style-type: none"> ● 采取噪音减少措施。 ● 复位 CPU 模块并重新运行。如果再次出现相同的错误, 这表示是一个 CPU 模块硬件故障。因此更换有故障的模块。也可以与最近的三菱代理商取得联系。 	○
1001			
1002			
1003			
1004			
1005			
1006			
1009	电源模块, CPU 模块, 主基板, 扩展基板故障或者检测到扩展电缆。	复位 CPU 模块并重新运行。如果再次出现相同的错误, 则电源模块, CPU 模块主基板, 扩展基板或扩展电缆出现故障。也可以与最近的三菱代理商取得联系。	○ 序列号 04122 或之后
1010	在没有执行 END 指令的情况下, 执行整个程序。 (1) 当 END 指令执行时, 例如由于噪音等原因而被读取成另外一个代码。 (2) END 指令不知何故已经被转变成另一个指令。	<ul style="list-style-type: none"> ● 采取噪音减少措施。 ● 复位 CPU 模块并再次运行。如果再次出现相同的错误, 这表示是一个 CPU 模块硬件的故障。因此更换有故障的模块。也可以与最近的三菱代理商取得联系。 	○
1011			
1012			
1020	一个 SFC 程序不能正常结束。 (1) SFC 程序由于噪音或其它原因不能正常结束。 (2) SFC 程序由于一些原因不能正常地结束。	<ul style="list-style-type: none"> ● 采取噪音减少措施。 ● 复位 CPU 模块并再次运行。如果再次出现相同的错误, 这表示是一个 CPU 模块硬件故障。因此更换有故障的模块。也可以与最近的三菱代理商取得联系。 	○ 序列号 04122 或之后
1101	存储 CPU 顺序程序的程序内存里出现错误。	这表示是一个 CPU 模块硬件故障。因此, 需要更换有故障的模块。也可以与最近的三菱代理商取得联系。	○
1102	RAM 里的错误被用于 CPU 工作区域。		
1103	内部 CPU 元件错误。		
1104	CPU 里的 RAM 地址错误。		
1105	CPU 模块里的 CPU 内存错误。		
1200	执行 CPU 内部索引修改的电路不能正常工作。	这表示是一个 CPU 模块硬件故障。因此, 需要更换有故障的模块。也可以与最近的三菱代理商取得联系。	○
1201	内部 CPU 硬件(逻辑)不能正常工作。		
1202	在 CPU 里执行顺序处理的电路不能正常工作。		
1300	有一个输出模块, 保险丝烧坏。	<ul style="list-style-type: none"> ● 检查输出模块的 FUSE LED, 同时更换 LED 点灯亮的模块。 ● 有保险丝烧坏的模块也可以用外围设备来检查。监视从 SD130 到 SD137 的特殊寄存器, 并检查是否有一个位是 1, 它对应保险丝烧坏模块。 	○ 序列号 04122 或之前的
1310	没有中断模块, 但中断出现了。	模式中的一个有硬件故障。因此, 检查模块同时更换有故障的模块。也可以与最近的三菱代理商取得联系。	
1311	检测到来自中断模块外的中断要求。	采取措施, 使中断不能由中断模块以外的模块引起。	○ 序列号 04122 或之前的
	检测到来自在 PLC 参数对话框中没有设置中断指针设定的模块的中断要求。	<ul style="list-style-type: none"> ● 修改在 PLC 参数对话框中的 PLC 系统设定中的中断指针设置。 ● 采取措施, 使得中断不是由在 PLC 参数对话框的 PLC 系统设定的中断指针设定没有设定的模块引起。 修改网络参数的中断设定。 修改智能功能模块缓冲内存的中断设定。 修改 QD51 的基本程序。 	
1401	<ul style="list-style-type: none"> ● 在初始通信阶段, 没有来自智能功能模块的响应。 ● 智能功能模块的缓冲内存大小是错误的。 	这表示是一个 CPU 模块的硬件故障。因此, 更换故障模块。也可以与最近的三菱代理商取得联系。	○
1402	在程序里访问了智能功能模块, 但是没有响应。	这表示是一个 CPU 模块的硬件故障。因此, 更换故障模块。也可以与最近的三菱代理商取得联系。	
1403	<ul style="list-style-type: none"> ● 当执行 END 指令时, 智能功能模块没有响应。 ● 在智能功能模块里一个错误被检测。 	被访问的智能功能模块有硬件故障。因此, 更换故障的模块。也可以与最近的三菱代理商取得联系。	

*1 圆括号里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
1411	CONTROL-BUS ERR.	单元号码	—	关	闪烁	停止	在电源打开/复位时	
1412			程序错误位置				FROM/T0 指令集的执行过程中。	
1413		—	—	关	闪烁	停止	一直	
1414		—	—	关	闪烁	停止	当 END 指令被执行时。	
1415		基本号码		关	闪烁	停止	当 END 指令执行时。	
1416		PLC 号码		关	闪烁	停止	在电源打开/复位时	
1500	AC/DC DOWN	—	—	开	关	继续	一直	
1600	BATTERY ERROR	驱动器名称	—	开	开	继续	一直	
2000	UNIT VERIFY ERR.	单元号码	—	关/开	闪烁/开	停止/继续 *2	当 END 指令执行时。	
2001	UNIT VERIFY ERR.	PLC 号码	—	关/开	闪烁/开	停止/继续	当 END 指令执行时。	
2010	BASE LAY ERROR.	基本号码	—	关	闪烁	停止	在电源打开/复位时	
2011								
2100	SP. UNIT LAY ERR.	单元号码	—	关	闪烁	停止	在电源打开/复位时	
2103								
2106								
2107								

*1 在圆括号里的字符表示存储个别信息的特殊寄存器的号码。

*2 错误出现时的 CPU 操作状态能在参数下设定。(LED 显现会相应地改变。)

错误代码 (SD0) *1	错误内容和原因	修改方法	相应 CPU
1411	当执行参数 I/O 分配时, 智能功能模块在初始通信时不能访问。在错误出现时, 相应特殊功能模块的头 I/O 号码存储在公共信息里。	复位 CPU 模块并再次运行。如果再次出现错误, 智能功能模块, CPU 模块或基板有故障。更换故障模块或单元。也可以与最近的三菱代理商取得联系。	○
1412	FROM/TO 指令由于智能功能模块的系统总线错误不能执行。在错误出现时, 程序错误位置被存储。		○
1413	在多 CPU 系统配置里, 功能版本 A 的 QCPU 被安装。	从主机板中去掉功能版本 A 的 QCPU。	序列号 04122 或之后
	在系统总线上检测到错误。 (1) 系统总线的自我诊断错误。 (2) CPU 模块的自我诊断错误。	复位 CPU 模块并再次运行。如果再次出现错误, 智能功能模块, CPU 模块或基板有故障。更换故障模块或单元。也可以与最近的三菱代理商取得联系。	○
1414	在多 CPU 系统配置里, 功能版本 A 的 QCPU 被安装。	从主基板中去除功能版本 A 的 QCPU。	序列号 04122 或之后
	在系统总线上检测到错误。	复位 CPU 模块并再次运行。如果再次出现错误, 智能功能模块, CPU 模块或基板有故障。更换故障模块或单元。也可以与最近的三菱代理商取得联系。	○
1415	主基板或扩展基板的故障被检测到。	复位 CPU 模块并再次运行。如果再次出现错误, 智能功能模块, CPU 模块或基板有故障。更换故障模块或单元。也可以与最近的三菱代理商取得联系。	○
1416	在多 CPU 系统配置里当电源打开或复位时, 总线故障被检测到。	复位 CPU 模块并再次运行。如果再次出现错误, 智能功能模块, CPU 模块或基板有故障。更换故障模块或单元。也可以与最近的三菱代理商取得联系。	序列号 04122 或之后
1500	电源出现瞬间电源中断。 电源消失。	检查电源。	○
1600	(1) 在 CPU 模块电池的电压降低到低于规定水平。 (2) CPU 模块电池的前连接器没有连接。	● 换电池。 ● 如果电池用于程序内存, 标准 RAM 或用于备份电源功能, 安装导线连接器。	○
2000	I/O 模块信息电源 ON 被改变。 (1) I/O 模块(或智能功能模块)没有正确安装或安装在基板上。	读取外围设备上的错误公共信息, 同时检查或改变和号码值对应(模块号码)的模块。 也可以使用 GX DEVELOPER, 监视 SD150 到 SD157 的特殊寄存器, 同时检查和替换其数据位为 1 的模块。	○
2001	在操作中, 在 CPU 模块空闲设置的插槽上, 安装一个模块。	操作中, 在 CPU 模块空闲设置的插槽上, 不要安装模块。	○
2010	(1) 安装了 5 块或更多块扩展基板。 (2) 当显示设备是总线连接时, CPU 模块被复位时显示设备的电源为 OFF。	● 卸下第五块之后的扩展基板。 ● 再次接通 PLC 和人机接口电源。	序列号 04122 或之后
2011	QA□B 或 QA1S□B 用作基本单元。	不能将 QA□B 或 QA1S□B 作为基本单元使用。	○
2100	(1) 在参数 I/O 分配设定中, 智能功能模块被分配到一个为 I/O 模块保留的位置或反之。 (2) 在参数 I/O 分配设定中, 一个不是 CPU (或没有) 的模块被分配到为 CPU 模块保留的位置或反之。 (3) 在 PLC 参数对话框的 I/O 分配设定中, 分配给智能功能模块的点数少于安装的模块的点数。	● 复位参数 I/O 分配设定, 使其符合智能功能模块和 CPU 模块的状态。	○
2103	2 个或更多中断模块被安装。	将中断模块减少到 1 个。	序列号 04122 或之后
	在中断指针没有设定的位置, 安装了 2 个或更多中断模块。	● 将中断模块减少到 1 个。 ● 使得中断指针设定为第二个中断模块和其后的模块。	序列号 04122 或之后
2106	(1) 2 个或更多 MELSECNET/H 被安装。 (2) 2 个或 Q 系列 ETHERNET 模块被安装。 (3) 3 个或 Q 系列 CC-LINK 模块被安装。 (4) 在 MELSECNET/H 的网络系统里存在相同的网络号或相同的站点号。	● 将 MELSECNET/H 减少为 1 或更少。 ● 将 Q 系列 ETHERNET 模块数减少为 1 或更少。 ● 将 Q 系列 CC-LINK 模块减少为 2 或更少。 ● 检查网络号和站点号。	○
2107	参数 I/O 分配设定中的起始 X/Y 也是另一个模块的起始 X/Y。	复位参数 I/O 设定, 使得符合特殊功能模块的实际状态。	○

*1 圆括号里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
2110	SP. UNIT ERROR	单元号码	程序错误位置	关/开	关/开	停止/继续 * 2	当执行指令时。	
2111								
2112	SP. UNIT ERROR	单元号码	程序错误位置	关/开	闪烁/开	停止/继续 * 2	当执行指令时。 STOP → RUN	
2114	SP. UNIT ERROR	单元号码	程序错误位置	关/开	闪烁/开	停止/继续	指令执行时。	
2115								
2116								
2117								
2120	SP. UNIT LAY ERR.	—	—	关	闪烁	停止	在电源打开/复位时	
2122								
2124								
2125	SP. UNIT LAY ERR.	单元号码	—	关	闪烁	停止	在电源打开/复位时	
2200	MISSING PARA.	驱动名称	—	关	闪烁	停止	在电源打开/复位时	
2210	BOOT ERROR	单元号码	—	关	闪烁	停止	在电源打开/复位时	
2400	FILE SET ERROR	文件名称/驱动名称	参数号	关	闪烁	停止	在电源打开/复位时	
2401								
2500	CAN'T EXE. PRG.	文件名称/驱动名称	—	关	闪烁	停止	在电源打开/复位时	
2501								
2502								
2503								
2504								
3000	PARAMETER ERROR	文件名称/驱动名称	参数号	关	闪烁	停止	在电源打开/复位时 STOP → RUN	
3001								

*1 在圆括号里的字符表示个别信息存储地方的特殊寄存器号码。

*2 错误出现时的 CPU 操作状态能在参数中设定。(LED 显现会相应地改变)

错误代码 (SD0) *1	错误内容和原因	修改方法	相应 CPU
2110	(1) 由指令集 FROM/T0 指定的位置不是智能功能模块。 (2) 被访问的智能功能模块出现故障。	<ul style="list-style-type: none"> ● 读取错误个别信息，然后检查和编辑和号码值(程序错误位置)对应的 FROM/T0 指令集。 ● 被访问的智能功能模块有硬件错误。咨询最近的服务中心，代理商和我们的分公司办事处，描述其症状。 	○
2111	由链接直接元件(J□□□)指定的位置不是网络模块。		
2112	(1) 由智能功能模块专用指令指定的位置不是一个智能功能模块或者也不是相关的智能功能模块。 (2) 在网络专用指令中指定的网络号码不存在,或继电器目标网络不存在。		
2114	对主机 CPU 进行设置的指令被另一 CPU 设置所使用。(指令不允许主机 CPU 被指定。)	用 GX DEVELOPER 来读取公共信息，同时检查和修改对应于此值(程序错误地址)的程序。	○ 序列号 04122 或之后
2115	对另一 CPU 进行设置的指令被主机 CPU 设置所使用。(指令不允许另一 CPU 被指定。)		
2116	由另一 CPU 控制的模块被不允许被别的 CPU 控制的模块指令来设定。		
2117	在多 CPU 系统专用指令里不能指定的 CPU 模块被指定。		
2120	QACB 或 QASCB 用作主基板。	使用 QCB 作为主基板。	○ 序列号 04122 或之后
2122	QASCB 安装到主基板。	安装 QCB 作为主基板。	
2124	(1) 模块被在安装 25 或更高的插槽(17 号或更高的插槽用于 Q00J) (2) 在基板分配设定指定的插槽数之后的插槽内安装了模块。 (3) 在实际 I/O 点之后的 I/O 点上安装了模块。 (4) 在实际 I/O 点的边界处安装了模块。	<ul style="list-style-type: none"> ● 将 25 号或之后的插槽中的模块去掉(14 号或之后的插槽用于 Q00J) ● 去除在基板分配设定中指定的插槽数之后的插槽上安装的模块。 ● 去除在实际 I/O 点之后安装的 I/O 点模块。 ● 对于占据点没有超过实际 I/O 点的模块，改变最后一个模块。 	○
	(5) 5 个或更多扩展基板被添加。(3 个基板用于 Q00J)	● 去除 5 个或更多扩展主基板。(3 个基板用于 Q00J)	○ 序列号 04122 或之后
2125	(1) 一个不能辨认的模块被安装。 (2) 智能功能模块没有响应。	<ul style="list-style-type: none"> ● 安装可使用模块。 ● 智能功能模块有硬件故障。因此，更换有故障的模块。或者联系最近的三菱代理商。 	○
2200	在程序内存里没有参数文件。	设定参数文件到程序内存里。	○
2210	引导文件内容不合适。	重新检查引导设置。	○ 序列号 04122 或之后
2400	在参数里的 PLC 文件设定中指定的文件找不到。	使用 GX DEVELOPER 读取错误的个别信息，检查以确定和参数号码的号码值的对应参数驱动名称和文件名称，并改正错误。 创建指定的文件。	○
2401	在参数 PLC RAS 设定故障历史区域内指定的文件不能被建立。	使用 GX DEVELOPER 读取错误的个别信息，检查以确定和参数号码的号码值的对应参数驱动名称和文件名称，并改正错误。 检查在内存卡中剩余的空间。	
2500	(1) 有一个程序文件，它使用的元件超出了在 PLC 参数元件设定里设定的范围。 (2) 在 PLC 参数设定改变后，只有参数写进 PLC。	(1) 用 GX DEVELOPER 读取公共信息，同时检查分配给与(文件名称)数值和参数元件设定相对应的程序文件的元件。例如，对元件分配进行修正。 (2) 如果 PLC 参数元件设定改变，成批写入参数和程序文件进入 PLC。	○
2501	(1) 有 3 个或更多的程序文件。 (2) 程序文件的名称与程序内容不符。	<ul style="list-style-type: none"> ● 删除不必要的程序文件。 ● 使程序名称与程序内容匹配。 	
2502	程序文件不是 QCPU/QNCPU 兼容的。或者，文件内容不是顺序程序的内容。	<ul style="list-style-type: none"> ● 检查程序版本是否是 * * *.QPG ● 检查文件内容确信它们是顺序程序。 	
2503	完全没有程序文件。 (驱动名称存储在公共信息中。)	<ul style="list-style-type: none"> ● 检查程序配置。 ● 检查参数和程序配置。 	
2504	有 2 个或更多 SFC 程序。	减少 SFC 程序为 1。	○ 序列号 04122 或之后
3000	计时器的时间限定设置，RUN-PAUSE 触点，公共指针号，一般数据处理，空插槽的数量，或系统中断设定的参数设定超出了能被 CPU 模块使用的范围。	<ul style="list-style-type: none"> ● 使用 GX DEVELOPER 读取错误的详细信息，检查对应于那些号码值(参数号码)的参数条目，在必要时进行修改。 ● 在修改参数设定后，错误仍然存在，可能的原因是 CPU 模块的程序内存有内存错误。因此，更换有故障的模块。或者，联系最近的三菱代理商。 	○
3001	参数内容已经被破坏。		

*1 在圆括号里的字符表示个别信息存储地方的特殊寄存器号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
3003	PARAMETER ERROR	文件名称/驱动名称	参数号	关闭	闪烁	停止	电源打开/复位时 STOP → RUN	
3004								
3012								
3013								
3100	LINK PARA. ERROR	文件名称/驱动名称	参数号	关闭	闪烁	停止	电源打开/复位时 STOP → RUN	
3101								
3102	LINK PARA. ERROR	文件名称/驱动名称	参数号	关闭	闪烁	停止	电源打开/复位时 STOP → RUN/当 END 指令执行时。	
3103	LINK PARA. ERROR	文件名称/驱动名称	参数号	关闭	闪烁	停止	电源打开/复位时 STOP → RUN	
3104								

*1 在圆括号里的字符表示个别信息被存储的地方的特殊寄存器号码。

错误代码 (SD0) *1	错误内容和原因	修改方法	相应 CPU
3003	在参数元件设定中设定的元件数量超过了 CPU 允许模块范围。	<ul style="list-style-type: none"> ● 使用 GX DEVELOPER 读取错误的细节信息，检查对应于那些号码值(参数号)的参数条目，同时在必要时修改。 ● 在修改参数设定后，错误仍然存在，可能的原因是 CPU 模块的程序内存有内存错误。因此，更换有故障的模块。或者，联系最近的三菱代理商。 	○
3004	参数文件同 QCPU/QNCPU 不兼容。 或者，文件内容不是参数。	<ul style="list-style-type: none"> ● 检查是否程序版本是 * * *.QPG ● 检查文件内容确信它们是参数。 	○
3012	在一个多 CPU 系统配置里，多 CPU 系统的参数设定不同于参考 CPU (1 号 CPU) 的参数。	使多 CPU 系统的设置与参考 CPU 的设定相匹配。	○
3013	在一个多 CPU 的系统里，多 CPU 自动刷新设置是下列中的一个。 (1) 当位元件被指定为刷新元件时，一个 16 倍数以外的号码被指定为刷新起始元件。 (2) 指定元件是能被指定的元件以外的一个。 (3) 传送点数量是一个奇数 (4) 传送点的总数大于刷新点的最大数量。	在多 CPU 自动刷新参数里，检查和修改下列内容。 <ul style="list-style-type: none"> ● 当指定一个位元件，指定 16 的倍数作为刷新起始元件。 ● 指定的元件能被指定为刷新元件。 ● 设定传送点的号码为偶数。 ● 设定传送点的总数在刷新点的最大数量范围内。 	○ 序列号 04122 或之后
3100	(1) 实际安装模块的数量不同于在 MELSECNET/H 的模块设定参数中指定的数量。 (2) 实际安装模块的起始 I/O 号码不同于在 MELSECNET/H 的网络参数中指定的号码。 (3) 在参数里的一些数据不能处理 (4) MELSECNE/H 的站类型在电源上电时被改变。(需要用 RESET-RUN 来改变站点类型)	<ul style="list-style-type: none"> ● 检查网络参数和安装状态，如果它们不同，匹配网络参数和安装状态。当网络参数被修改时，写入 CPU 模块。 ● 检查扩展基板的扩展段号码的设定。 ● 检查扩展基板和连接器的连接状态。当显示设备连接到主基板或扩展基板时，检查显示设备的连接状态。 ● 如果按如上检查后，仍然有错误，这表示是硬件故障。因此，更换有故障的硬件。或者联系最近的三菱的代理商。 	○
	在一个多 CPU 系统组成里，由另一 CPU 控制的 MELSECNET/H 模块在 MELSECNET/H 网络设定参数中被指定为起始 I/O 号码。	删除由另一 CPU 控制的 MELSECNET/H 模块的网络设定参数。 改变由本机 CPU 控制的 MELSECNET/H 模块的起始 I/O 号码。	○ 序列号 04122 或之后
3101	(1) 在参数中指定的起始 I/O 号码和安装的模块中的一个不同。 (2) 在参数中指定的网络类型和模块中的一个不同。 MELSECNET/H 网络刷新参数在范围之外。	<ul style="list-style-type: none"> ● 检查网络参数和安装状态，如果它们不同，匹配网络参数和安装状态。当网络参数被修改时，写入 CPU 模块。 ● 检查扩展基板的扩展段号码的设定。 ● 检查扩展基板和连接器的连接状态。当显示设备连接到主基板或扩展基板时，检查显示设备的连接状态。 ● 如果按如上检查后，仍然有错误，这表示是硬件故障。因此，更换有故障的硬件。或者联系最近的三菱的代理商。 	○
3102	在网络模块里执行网络参数检查时发现一个错误。 指定给 MELSECNET/H 的参数不正常。	<ul style="list-style-type: none"> ● 修改并写入网络参数。 ● 如果在修改后仍然出现错误，这表示是硬件故障。因此，更换有故障的硬件。或者联系最近的三菱的代理商。 	○
3103	(1) 尽管在参数里以太网模块的数量设定为 1 个或更多，实际安装的模块数为 0。 (2) 以太网模块设定参数的起始 I/O 号码不同于实际安装的 I/O 模块的号码。	<ul style="list-style-type: none"> ● 修改并写入网络参数。 ● 如果在修改后仍然出现错误，这表示是硬件故障。因此，更换有故障的硬件。或者联系最近的三菱的代理商。 	○
	在多 CPU 系统里，被另一 CPU 控制的以太网模块被指定为以太网设定参数的起始 I/O 号码。	<ul style="list-style-type: none"> ● 删除由另一 CPU 控制的以太网模块的以太网模块参数。 ● 对于由本机 CPU 控制的以太网的起始 I/O 号码需要变化。 	Q00/Q01 序列号 04122 或之后
3104	(1) 以太网和 MELSECNET/H 使用同样的网络号码。由参数设定的网络号码，站点号码和组号码超出了范围。 (2) 指定的 I/O 号码超出了使用的 CPU 模块的范围， (3) 以太网指定参数设定不正常。	<ul style="list-style-type: none"> ● 修改和写入网络参数。 ● 如果在修改后仍然出现错误，这表示是硬件故障。因此，更换有故障的硬件。或者联系最近的三菱的代理商。 	○

*1 在圆括号里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
3105	LINK PARA. ERROR	文件名称/驱动名称	参数号	关闭	闪烁	停止	当电源打开/复位时 S10P→RUN	
3106 3107	LINK PARA. ERROR	文件名称/驱动名称	参数号	关闭	闪烁	停止	当电源打开/复位时 S10P→RUN	
3200	SFC. PARA. ERROR	文件名称/驱动名称	参数号	关闭	闪烁	停止	STOP→RUN	
3300 3301 3302 3303	SP. PARA. ERROR	文件名称/驱动名称 文件名称/驱动名称	参数号	关闭	闪烁	停止	当电源打开/复位时 S10P→RUN	
3400 3401	REMOTE PASS. ERROR	—	—	关闭	闪烁	停止	当电源打开/复位时 S10P→RUN	
4000 4001 4002 4003 4004 4010 4021 4030 4100 4101 4102 4103 4107 4111 4112	INSTRCT CODE ERR. MISSING END INS. CAN'T SET (P) CAN'T SET (I)	程序错误位置 程序错误位置	— —	关闭 关闭	闪烁 闪烁/打开	停止 停止/继续 * 2	当电源打开/复位时 S10P→RUN 当指令执行时。	

*1 在圆括号里的字符表示存储个别信息的特殊寄存器的号码。

*2 错误出现时的 CPU 操作状态能在参数下设定。(LED 显现会相应地改变)

错误代码 (SD0) *1	错误内容和原因	修改方法	相应 CPU
3105	(1) 尽管通过参数设定的 CC-Link 模块的数量是 1 或更多，模块的实际安装数量是 0。 (2) 公共参数的起始 I/O 号码不同于实际安装的 I/O 模块的号码。 (3) CC-Link 单元数量设置参数的站点类型不同于实际安装站点的站点类别。 在一个多 CPU 系统配置里，由另一 CPU 控制的 CC-Link 模块在 CC-Link 网络设定参数被指定为起始 I/O 号码。	<ul style="list-style-type: none"> ● 修改和写入网络参数。 ● 如果在修改后仍然出现错误，这表示是硬件故障。因此，更换有故障的硬件或者联系最近的三菱的代理商。 	○
3106	CC-Link 的网络刷新参数超过范围。	检查参数设定。	○
3107	CC-Link 参数的内容不正确	检查参数设定。	○
3200	参数设定非法。 (1) 尽管在 PLC 参数对话框的 SFC 设定中将块 0 设定为“自动启动”，但块 0 不存在。	使用 GX DEVELOPER 读取错误的公共性信息，同时检查和修改对应于该值(程序错误位置)的错误步骤。	○ 序列号 04122 或之后
3300	设定在 GX 组成器上的智能功能模块参数的起始 I/O 号码不同于实际 I/O 号码。	检查参数设定。	○
3301	智能功能模块的刷新参数设定超出了现有的范围。		
3302	智能功能模块的参数不正常。		
3303	在一个多 CPU 系统，自动刷新设定或其它一些参数设定被设定给另一 CPU 控制的智能功能模块。	<ul style="list-style-type: none"> ● 删除由另一 CPU 控制的智能功能模块的自动刷新设定的参数设定。 ● 改变由本机 CPU 控制的智能功能模块的自动刷新设定的参数设定。 	○ 序列号 04122 或之后
3400	远程口令的目标模块的起始 I/O 号码超出了下列范围： (1) Q00JCPU:0H1E0H (2) Q00CPU/Q01CPU:0H 到 3E0H	改变远程口令的目标模块的起始 I/O 号码为如下范围内： <ul style="list-style-type: none"> ● Q00JCPU:0H 到 1E0H ● Q00CPU/Q01CPU:0H 到 3E0H 	○
3401	对于远程口令的起始 I/O 号码，任何下列模式在指定的插槽上没有安装。 (1) 功能版本为 B 的串行通信模块。 (2) 功能版本为 B 的以太网模块。 (3) 功能版本为 B 的调制解调器接口模块。	对于远程口令的起始 I/O 号码指定的位置安装下列任一如下模块。 <ul style="list-style-type: none"> ● 功能版本为 B 的串行通信模块。 ● 功能版本为 B 的以太网模块。 ● 功能版本为 B 的调制解调器接口模块。 	○ 序列号 04122 或之后
4000	(1) 程序包含一个不能解码的指令代码。 (2) 在程序里包含一个不能使用的指令。	使用 GX DEVELOPER 读取错误的公共性信息，同时检查和修改对应于该值(程序错误位置)的错误步骤。	○
4001	尽管程序不是一个 SFC 程序，它包括 SFC 专用指令。		○ 序列号 04122 或之后
4002	(1) 由程序指定的扩展指令有个不正确的指令名称。		○
4003	(2) 由程序指定的扩展指令不能由指定的模块执行。		
4004	由程序指定的扩展指令有一个不正确的元件号码。		
4010	由程序指定的扩展指令指派了一个不能使用的元件。		
4021	在程序里没有 END 指令。	使用 GX DEVELOPER 读取错误的公共性信息，同时检查和修改对应于该值(程序错误位置)的错误步骤。	○
4030	单个文件使用的公共指针重叠。		
4100	单个文件使用的位置指针号码重叠。		
4101	包括无法通过指令进行处理的数据	使用 GX DEVELOPER 读取错误的公共性信息，同时检查和修改对应于该值(程序错误位置)的错误步骤。	○
4102	(1) 指令处理的数据的指定元件号码超出了使用范围。 (2) 指令指定的元件存储数据或常量超出了使用范围。 (1) 专用网络指令指定的网络号码和站点号码不正确。 (2) 链接直接元件(└┐┐┐┐┐)没被正确设定。 在多 CPU 系统里，为网络模块指定的链接直接元件(└┐┐┐┐┐)被其他 CPU 控制。		
4103	PID 专用指令的结构不正确。	<ul style="list-style-type: none"> ● 删除从程序里由另一 CPU 控制的网络模块被指定的链接直接元件。 ● 对于被本站控制的网络模块的链接直接元件地指定。 	○ 序列号 04122 或之后
4107	33 个更多的多 CPU 专用指令从一个 CPU 模块执行。	使用 GX DEVELOPER 读取错误的公共性信息，同时检查和修改对应于该值(程序错误位置)的错误步骤。	○ 序列号 04122 或之后
4111	用一个指令尝试写入到本机 CPU 模块的 CPU 共享内存不能写入区域。	使用 GX DEVELOPER 读取错误的公共性信息，同时检查和修改对应于该值(程序错误位置)的错误步骤。	○ 序列号 04122 或之后
4112	不能用多 CPU 专用指令指定的 CPU 模块被指定。		

*1 在圆括号里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
4200	FOR NEXT ERROR	程序错误位置	—	关闭	关闭	停止	当指令执行时	
4201								
4202								
4203								
4210	CAN' T EXECUTE (P)	程序错误位置	—	关闭	关闭	停止	当指令执行时	
4211								
4212								
4213								
4220	CAN' T EXECUTE (P)	程序错误位置	—	关闭	关闭	停止	当指令执行时	
4221	CAN' T EXECUTE (I)	—	—	关闭	关闭	停止	当指令执行时	
4222	CAN' T EXECUTE (I)	程序错误位置	—	关闭	关闭	停止	当指令执行时	
4223	CAN' T EXECUTE (I)	—	—	关闭	关闭	停止	当指令执行时	
4231	INST. FORMAT ERR.	程序错误位置	—	关闭	关闭	停止	当指令执行时	
4410	CAN' T SET (BL)	程序错误位置	—	关闭	关闭	停止	STOP → RUN	
4411								
4420	CAN' T SET (S)	程序错误位置	—	关闭	关闭	停止	STOP → RUN	
4421								
4422								
4423								
4430	SFC EXE. ERROR	程序错误位置	—	关闭	关闭	停止	STOP → RUN	
4431								
4432								
4502	SFCP. FORMAT ERR.	程序错误位置	—	关闭	关闭	停止	STOP → RUN	
4503								
4504								
4505								
4506								
4621	BLOCK EXE. ERROR	程序错误位置	—	关闭	关闭	停止	当指令执行时	
4631	STEP EXE. ERROR	程序错误位置	—	关闭	关闭	停止	当指令执行时	

*1 在圆括号里的字符表示存储个别信息的特殊寄存器的号码。

错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
4200	在 FOR 指令执行后没有执行 NEXT 指令。或者 NEXT 指令少于 FOR 指令。	使用 GX DEVELOPER 读取错误的公共信息，同时检查和修改对应于该值(程序错误位置)的错误步骤。	○
4201	尽管没有 FOR 指令被执行，一个 NEXT 指令被执行。或者 NEXT 指令多于 FOR 指令。		
4202	程序有多于 16 级的嵌套。		
4203	尽管在之前没有 FOR 指令被执行，一个 BREAK 指令执行。		
4210	CALL 指令执行，但是在指定指针处没有子函数。		
4211	在执行子函数的程序里没有 RET 指令。		
4212	在主程序里，RET 指令在 FEND 指令之前。	保持嵌套级别在 16 或更低。	○
4213	插入的嵌套级数超出 16。		
4220	尽管发生输入中断情况，却无相应的中断指针。		
4221	在所执行的中断程序中没有 IRET 指令。	使用 GX DEVELOPER 读取错误的公共信息，同时检查和修改对应于该值(程序错误位置)的错误步骤。	○
4223	在主程序中，IRET 指令在 FEND 指令之前。		
4231	IX 和 IXEND 指令的号码不等。		
4410	在 SFC 程序里指定的块号码超过最大值。	使用 GX DEVELOPER 读取错误的公共信息，同时检查和修改对应于该值(程序错误位置)的错误步骤。	○
4411	同样的块号码在 SFC 程序的两个或更多地方被指定。		
4420	在 SFC 程序里指定的步号码超过了最大值。	减少步的总数到或小于最大值。	○
4421	在 SFC 程序里指定的总步数超过了最大值。		
4422	在 SFC 程序里指定的步号码超过了最大值。	使用 GX DEVELOPER 读取错误的公共信息，同时检查和修改对应于该值(程序错误位置)的错误步骤。	○
4423	步继电器在 SFC 程序里不足。		
4430	SFC 程序不能执行。 (1) 块数据设定的数据不合法。 (2) 块数据设定的 SFC 数据元件超过了在 PLC 参数里设定的元件设定范围。	● 使用 GX DEVELOPER 再次写入程序到 CPU 模块 ● 在修改 SFC 数据元件的设定后，写入到 CPU 模块。 ● 在 PLC 参数设定里修改元件设定范围后，写入到 CPU 模块	○
4431	SFC 程序不能执行。 (1) 块参数设定不正常。		
4432	SFC 程序不能执行。 (1) SFC 程序的结构不合法。		
4502	SFC 程序的结构不合法。 (1) 在 SFC 程序块里不存在 STEP*指令。	使用 GX DEVELOPER 再次写入程序到 CPU 模块。	○
4503	SFC 程序的结构不合法。 (1) 在 TSET 指令指定的步不存在。 (2) 在跳越转移时，主机步号码被指定为目标步号码。		
4504	SFC 程序的结构非法。 (1) 在 TAND 指令中指定的步不存在。	使用 GX DEVELOPER 再次写入程序到 CPU 模块。	○
4505	SFC 程序的结构非法。 (1) 在一步的操作输出中，SET SNBLMSN 或 RSTSNBLMSN 指令为主机位指定。		
4506	SFC 程序的结构非法。 (1) 在复位步骤时，主机步号码被指定为目标步骤。		
4621	在 SFC 程序里不存在的块被启动	● 使用 GX DEVELOPER 读取错误的公共信息检查和修改对应于该值(程序错误位置)的错误步。 ● 如果 SM321 关闭，打开它。	○
4631	在 SFC 程序里不存在的步被启动。		

*1 在圆括号里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
5001	WDT ERROR	时间(设定值)	时间(实际测量值)	关闭	闪烁	停止	一直	
5010	PRG. TIME OVER	时间(设定值)	时间(实际测量值)	打开	打开	继续	一直	
7000	MULTI CPU DOWN	CPU 号码.	—	关闭	闪烁	停止	一直	
							电源打开/复位时	
7002	MULTI CPU DOWN	CPU 号码	—	关闭	闪烁	停止	电源打开/复位时	
7003								
7004	MULTI CPU DOWN	CPU 号码	—	关闭	闪烁	停止	一直	
7010	MULTI EXE. ERROR	CPU 号码	—	关闭	闪烁	停止	电源打开/复位时	
7011								
7020	MULTI CPU. ERROR	CPU 号码	—	打开	打开	继续	一直	
7030	CPU LAY ERROR	CPU 号码	—	关闭	闪烁	停止	电源打开/复位时	
7031								
7032								
7035	CPU LAY ERROR	单元号码	—	关闭	闪烁	停止	电源打开/复位时	
9000	F**** *3	程序错误位置	报警器号码	打开	打开/关闭 *4	继续	当指令执行时	

*1 在圆括号里的字符表示存储个别信息的特殊寄存器的号码。

*3 ****指示检测到的报警器号码。

*4 可以被用于 LED 显示优先级设定的特殊寄存器(SD207 到 SD209)保持为关闭。

错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
5001	程序扫描时间超过了在 PLC RAS 设定里的 WDT 值。	使用 GX DEVELOPER 读取错误的个别信息同时检查号码值, 尽可能较少扫描时间。	○
5010	程序的扫描时间超过了在 PLC RAS 设定参数中指定的固定扫描设定时间。	检查固定扫描的设定时间。	
7000	(1) 在多 CPU 系统配置里, CPU 故障出现在操作模式的“系统停止被选择”的 CPU 号码的 CPU 模块里。 (2) 在多 CPU 系统配置里, 多 CPU 系统不兼容的 CPU 模块被安装。 在一个多 CPU 系统里, 在 CPU 的电源上电时第 1 号 CPU 出现一个停止错误, 其它 CPU 也不能启动。(这个错误在 2 号和 3 号 CPU 中也出现。)	<ul style="list-style-type: none"> ● 读取错误的个别信息, 确认导致 CPU 故障的 CPU 模块错误, 同时去除错误。 ● 从主基板中卸除多 CPU 系统不兼容的 CPU 模块。 	○ 序列号 04122 或之后
7002	(1) 在一个多 CPU 系统配置的初始化通信的时候, 从初始化通信目标的 CPU 号码没有返回任何响应。 (2) 在一个多 CPU 系统的构成里, 一个多 CPU 系统不兼容的 CPU 模块被安装。	<ul style="list-style-type: none"> ● 复位 CPU 模块并再运行, ● 如果再次出现相同的错误, 这表示有 CPU 模块的硬件故障。因此, 更换有故障的硬件。或者联系最近的三菱的代理商。 ● 从主基板中卸存多 CPU 系统不兼容的 CPU 模块。 	
7003	一个多 CPU 系统配置的初始化通信的时候, 初始化通信目标的 CPU 号码没有返回任何响应。	<ul style="list-style-type: none"> ● 复位 CPU 模块并再运行, ● 如果再次出现相同的错误, 这表示有 CPU 模块的硬件故障。因此, 更换有故障的硬件。或者联系最近的三菱的代理商。 	
7004	在一个多 CPU 系统配置里, 在 CPU 模块的通信之间出现一个数据错误	<ul style="list-style-type: none"> ● 检查在系统配置里超过 I/O 点的最大号码的安装模块。 ● 当在系统配置里没有问题, 这表示是 CPU 模块的硬件故障。因此, 更换故障模块。或者, 联系最近的三菱的代理商。 	
7010	(1) 在一个多 CPU 系统配置里, 有故障的 CPU 模块被安装。 (2) 在一个多 CPU 系统配置里, 一个多 CPU 系统不兼容的 CPU 模块被安装。(在序列号为 04122 或之后的 Q00CPU 和 Q01CPU 中检测到一个错误。) (3) 在一个多 CPU 系统里, 2 号和 3 号 CPU 在电源上电时复位。(在复位取消处的 CPU 号码里检测到错误。) (4) 在一个多 CPU 系统里, 版本为 1.06 或之前的 QFB(总线接口驱动)与 PC CPU 模块一起使用。	<ul style="list-style-type: none"> ● 使用 GX DEVELOPER 读取错误的个别信息同时替换故障 CPU 号码的 CPU 模块。 ● 用多 CPU 系统的兼容 CPU 模块来替换 CPU 模块。 ● 不要复位 2 号和 3 号 CPU 的 CPU 模块。 ● 改变版本是 QFB 和更迟的 PC 的 CPU 模块。 ● 复位 1 号 CPU 模块同时再次启动多 CPU 系统。 	○ 序列号 04122 或之后
7011	在一个多 CPU 的系统配置里, 以下的任何一个设定已经建立 (1) 对于在多 CPU 中不能执行自动刷新的 CPU 模块在多 CPU 自动刷新设定中被指定。 (2) 非组输入设定被指定给不能执行非组输入的 CPU 模块里。	<ul style="list-style-type: none"> ● 修改多 CPU 自动刷新设定。 ● 修改非组输入设定。 	
7020	在多 CPU 系统的构成里, 一个 CPU 故障出现在操作模式为“系统停止被选择”的 CPU 号码的 CPU 模块里。(在 CPU 故障没有出现的 CPU 模块里检测到错误。)	使用 GX DEVELOPER 读取错误的个别信息确认导致 CPU 故障的 CPU 模块的错误。	
7030	一个分配错误出现在某一可安装的插槽(CPU 插槽, I/O 插槽 0, 1)里, 此插槽安装的 CPU 模块超过了在 PLC 参数对话框的多 CPU 设定中设定的 CPU 模块数。	<ul style="list-style-type: none"> ● 用 PLC 参数对话框里设定的多 CPU 设定里的 CPU 模块号码匹配安装 CPU 模块(包括 CPU(空))的号码。 ● 与安装 CPU 模块的类型来匹配在 PLC 参数对话框里 I/O 分配的设定的类型。 	○ 序列号 04122 或之后
7031	在 PLC 参数对话框的多 CPU 设定的 CPU 号码的范围里, 出现一个分配错误。		
7032	(1) 安装的组成多 CPU 系统的 CPU 模块的数量是错误的。(Q00CPU 或 Q01CPU: 多于一个模块, 运行 CPU: 多于一个模块, PC CPU 模块: 多于一个模块) (2) 在 2 号 CPU 或靠后的 CPU 的插槽上安装 PC CPU 模块。 (3) 在 3 号 CPU 的插槽上安装运动 CPU。	每个 Q00CPU 或 Q01CPU, 运动 CPU 和 PC CPU 模块不应该超过 1。	
7035	CPU 模块安装在不能安装 CPU 模块的插槽上。	从 CPU 插槽, I/O 插槽 0, 1, 之外的插槽上卸载 CPU 模块,	○
9000	警报器 F 变为 ON。	使用 GX DEVELOPER 读取错误的个别信息, 检查对应于号码值(警报器号码)的程序。	

* 1 在圆括号里的字符表示存储个别信息的特殊寄存器的号码。

11.2.2 高性能型 QCPU/QNACPU 的错误列表

下列信息涉及错误代码，错误信息意义，原因和修改的措施。

在相应 CPU 栏目的“○”表示错误适用于各种类型的 CPU。“REM”表示同远程 I/O 模块兼容。在栏里的 CPU 类型名称表示错误只适用于专用的 CPU 类型。

错误代码 (SD0) * 1	错误信息	公共信息 (SD5 到 15) * 1	个别信息 (SD16 到 26) * 1	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
1000	MAIN CPU DOWN	—	—	关闭	闪烁/打开	停止	一直	
1001								
1002								
1003								
1004								
1005								
1006				关闭	闪烁	停止	一直	
1007								
1008								
1009								
1010	END NOT EXECUTE	—	—	关闭	闪烁	停止	当指令执行时	
1011								
1012								
1101	RAM ERROR	—	—	关闭	闪烁	停止	在电源打开/复位时	
1102								
1103								
1104								
1105								
1200	OPE. CIRCUIT ERR.	—	—	关闭	闪烁	停止	在电源打开/复位时	
1201								
1202								
1203								
1204							当指令执行时	
1205								
1206							当指令执行时	
1300	FUSE BREAK OFF	单元号码	—	关闭/打开	闪烁/打开	停止/继续 * 2	当指令执行时	

* 1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

* 2 错误出现时的 CPU MODULE 的操作状态能在参数下设定。(LED 显现会相应地改变)

错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
1000	运行模式挂起或主 CPU 的故障。 (1) 由于噪音和其它原因引起的功能故障。 (2) 硬件故障。	<ul style="list-style-type: none"> ● 采取噪音减少措施。 ● 复位 CPU 模块并再次运行。如果再次出现相同的错误, 这表示是一个 CPU 模块硬件故障。因此更换有故障的模块。也可以与最近的三菱代理商取得联系。 	○
1001			QCPU
1002			
1003			
1004			QCPU Rem
1005			
1006			QCPU
1007			
1008			
1009	电源模块, CPU 模块, 主基板, 扩展基板或者扩展电缆被检测到故障。	复位 CPU 模块并重新运行。如果再次出现相同的错误, 电源模块, CPU 模块主基板, 扩展基板或扩展电缆有故障。更换有故障的模块, 单元或电缆。也可以与最近的三菱代理商取得联系。	序列号 04101 或 之后
1010	在没有执行 END 指令的情况下, 执行整个程序。 (1) 当 END 指令执行时, 例如由于噪音被读取成另外一个代码。 (2) END 指令不知何故已经被转变成另一个指令。	<ul style="list-style-type: none"> ● 采取噪音减少措施。 ● 复位 CPU 模块并再次运行。如果再次出现相同的错误, 这表示是一个 CPU 模块的硬件故障。因此更换有故障的模块。也可以与最近的三菱代理商取得联系。 	○
1011			
1012			
1101	存储 CPU 顺序程序的内部 RAM/程序内存里出错。	这表示一个 CPU 模块硬件故障。因此, 需要更换有故障的模块。也可以与最近的三菱代理商取得联系。	○
1102	RAM 里的错误用作 CPU 模块的工作区域。		
1103	内部 CPU 模块元件内存错误。		
1104	CPU 模块里的 RAM 地址错误。		
1105	CPU 模块的系统 RAM 故障。	这表示一个 CPU 模块硬件故障。因此, 需要更换有故障的模块。也可以与最近的三菱代理商取得联系。	Q4AR
	CPU 模块的 CPU 共享内存故障。		
1200	执行 CPU 内部索引的电路不能正常运行。	这表示一个 CPU 模块硬件故障。因此, 需要更换有故障的模块。也可以与最近的三菱代理商取得联系。	○
1201	内部 CPU 模块硬件(逻辑)不能正常运行。		
1202	CPU 模块中执行顺序处理的电路工作不正常。		
1203	在 CPU 模块中进行索引修改工作的运算电路工作不正常。	这表示一个 CPU 模块硬件故障。因此, 需要更换有故障的模块。也可以与最近的三菱代理商取得联系。	Q4AR
1204	在 CPU 的硬件(逻辑)不能正常工作。		
1205	在 CPU 里执行顺序处理的操作电路不能正常工作。		
1206	在 CPU 里的 DSP 操作电路不能正常操作。		
1300	有一个保险丝烧坏的输出模块。	<ul style="list-style-type: none"> ● 检查输出模块的 FUSE LED, 同时当 LED 点亮时, 替换模块。 ● 使用外围设备来读取公共信息, 同时在对应该于数字值(模块号)的输出模块里更换保险丝。 或者, 用外围设备监视特殊寄存器 SD1300 到 SD1331, 同时改变位里有 1 值的输出模块的保险丝。 ● 当显示元件连接到主基板或扩展基板, 检查扩展电缆的连接状态和显示元件的接地状态。 	QCPU Rem
	有一个保险丝烧坏的输出模块。	<ul style="list-style-type: none"> ● 检查输出模块的 FUSE LED, 同时当 LED 点亮时, 替换模块。 ● 使用外围设备来读取公共信息, 同时在对应该于数字值(模块号)的输出模块里更换保险丝。 或者, 用外围设备监视特殊寄存器 SD1300 到 SD1331, 同时改变位里有 1 值的输出模块的保险丝。 ● 当显示元件连接到主基板或扩展基板, 检查扩展电缆的连接状态和显示元件的接地状态。 	QnA Q4AR
	(1) 有一个保险丝烧坏的输出模块。 (2) 输出装载的外部电源被关闭或未连接。	<ul style="list-style-type: none"> ● 检查输出模块的 FUSE LED, 同时当 LED 点亮时, 替换模块。 ● 使用外围设备来读取公共信息, 同时在对应该于数字值(模块号)的输出模块里替换保险丝, 或者用外围设备监视特殊寄存器 SD1300 到 SD1331, 同时改变位里有 1 值的输出模块的保险丝。 ● 检查用于外部装载的外部电源是 ON 或 OFF。 ● 当显示元件连接到主基板或扩展基板, 检查扩展电缆的连接状态和显示元件的接地状态。 	Q2AS

* 1 圆括号 () 里的字符表示个别信息存放地方的特殊寄存器号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
1310	I/O INT ERROR	—	—	关闭	闪烁	停止	在中断时	
1401	SP. UNIT DOWN	单元号码	—	关闭/打开	闪烁/打开	停止/继续 *3	电源打开/复位/当智能功能模块 被访问时	
1402	SP. UNIT DOWN	单元号码	程序错误位置	关闭/打开	闪烁/打开	停止/继续 *6	当智能功能模块访问指令执行时	
1403			—				在 FROM/TO 指令集执行的过程	
1411	CONTROL-BUS ERR.	单元号码	—	关闭	闪烁	停止	在电源打开/复位时	
1412			程序错误位置				在 FROM/TO 指令集执行的过程	
1413	CONTROL-BUS. ERR.	—	—	关闭	闪烁	停止	一直	
1414	CONTROL-BUS. ERR.	单元号码	—	关闭	闪烁	停止	当执行 END 指令时	
		—						
1415	CONTROL-BUS. ERR.	基本号码	—	关闭	闪烁	停止	当执行 END 指令时	
1416		单元号码					在电源打开/复位时	
1421	SYS. UNIT DOWN *3	—	—	关闭	闪烁	停止	一直	
1500	AC/DC DOWN	—	—	打开	关闭	继续	一直	
1510	DUAL DC DOWN 5V *4	—	—	打开	打开	继续	一直	
1520	DC DOWN 5V *5	—	—	关闭	闪烁	停止	一直	
1530	DC DOWN 24V *3	—	—	打开	打开	继续	一直	
1600	BATTERY ERROR	驱动号码	—	打开	打开	继续	一直	
1601				BAT. ALM LED On				
1602								

*1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

*3 只有在冗余系统里能够被检测。检测在控制系统和备用系统都可能。

*4 只有在冗余系统控制里可以被检测。

*5 在备用系统或冗余系统里可以检测。然而，在冗余系统里只能在控制系统里被检测。

*6 对于通过设定参数的每个模块，停止/继续操作是可选择的。

错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
1310	尽管没有中断模块，出现了中断。	安装模块之一有硬件故障。因此，检查安装模块并更换有故障的模块。或者，联系最近的三菱代理商。	○
1401	(1) 在初始通信阶段智能功能模块没有响应。 (2) 智能功能模块的缓冲内存的大小出错。 参数 I/O 分配已经建立，在初始处理阶段特殊功能模块没有返回信号。当发生错误时，和公共信息相对应的特殊功能模块的头 I/O 号码被存储。	这表示一个 CPU 模块硬件故障。因此，更换有故障的模块。或者，联系最近的三菱代理商。 被访问的特殊功能模块有硬件故障，因此，更换有故障的模块。或者，联系最近的三菱代理商。	QCPU Rem QnA
1402	特殊智能功能模块在程序里被访问，但是没有任何响应。 在执行 FROM/TO 指令集的过程中，特殊功能模块被访问，但是没有响应。当错误出现时，对应于个别信息的程序错误位置被存储。	这表示一个 CPU 模块硬件故障。因此，更换有故障的模块。或者，联系最近的三菱代理商。 被访问的特殊功能模块有硬件错误。联系最近的三菱代理商。	QCPU Rem QnA
1403	(1) 当执行 END 指令时，智能功能模块没有响应。 (2) 在智能功能模块上检测到错误。	被访问的特殊功能模块有硬件故障，因此，更换有故障的模块。或者，联系最近的三菱代理商。	QCPU Rem
1411	当执行一个参数 I/O 分配时，在初始通信中，不能访问特殊功能模块。在错误出现时，对应于特殊功能模块的起始 I/O 号码存储在公共信息里。	复位 CPU 模块并再运行。如果出现相同的错误，智能功能模块，特殊功能模块 CPU 模块或基板有故障。因此，更换有故障的模块或单元。或者，可以联系最近的三菱代理商。	○ Rem
1412	由于有特殊功能模块的系统总线错误，FROM/TO 指令集不能执行。在错误出现时，程序错误位置存储在个别信息里。		○
1413	在多 CPU 系统组态里，功能版本为 A。 在系统总线上检测到错误。 (1) 系统总线的自我诊断错误。 (2) CPU 模块的自我诊断错误。	● 从主基板中去除功能版本为 A 的高性能型 QCPU。或者，用功能版本为 B 的高性能型 QCPU 来替换功能版本为 A 的高性能型 QCPU。 ● 智能功能模块，CPU 模块或基板有故障。因此，更换有故障的模块或单元。或者，联系最近的三菱代理商。 复位 CPU 模块并再运行。如果出现相同的错误，智能功能模块，特殊功能模块 CPU 模块或基板有故障。因此，更换有故障的模块或单元。或者，可以联系最近的三菱代理商。	功能版本为 B 或以后的 QCPU QCPU Rem
1414	(1) 装载模块的故障被检测。 (2) 在多 PLC 系统组态里，功能版本为 B 的高性能型 QCPU 被安装。 在系统总线上检测一个错误。	● 从主基板中去除功能版本为 A 的高性能型 QCPU。或者，用功能版本为 B 的高性能型 QCPU 来替换功能版本为 A 的高性能型 QCPU。 ● 复位 CPU 模块并再运行。如果出现相同的错误，智能功能模块/特殊功能模块 CPU 模块或基板有故障。因此，更换有故障的模块或单元。或者，可以联系最近的三菱代理商。 特殊功能模块，CPU 模块或基板有故障。因此，更换有故障的模块或单元。或者，联系最近的三菱代理商。	功能版本为 B 或以后的 QCPU QCPU Rem
1415	主基板和扩展基板检测到故障。	特殊功能模块，CPU 模块或基板有故障。因此，更换有故障的模块或单元。或者，联系最近的三菱代理商。	功能版本为 B 或以后的 QCPU
1416	在 PLC 电源打开或 CPU 模块复位时系统总线的错误被检测到。		
1421	系统管理模块 AS92R 的硬件故障。	● 这表示是系统管理模块 AS92R 的硬件错误。因此，更换有故障的模块或单元。或者，联系最近的三菱代理商。	Q4AR
1500	(1) 瞬间电源中断出现。 (2) 电源关闭。	检查电源。	○ Rem
1510	电源转接扩展基板单元上的两个电源模块的任何一个电源电压(100 - 240VAC)降低到或少于额定电压的 85%。	检查电源模块的供给电压。如果电压故障，需要替换电源模块。	Q4AR
1520	电源转接扩展基板单元上的两个电源模块的任何一个电源电压(100 - 240VAC)降低到或少于额定电压的 85%。		
1530	供给系统管理模块的 AS92R 的 24VDC 电源降低到低于额定电压的 90%。	检查系统管理模块 AS92R 的 24VDC 电源。	
1600	(1) CPU 模块电池的电压降低到规定水平。 (2) CPU 模块电池没有连接。	● 更换电池 ● 如果电池用于程序内存，标准 RAM 或备份电源功能，安装前连接器。	○
1601	在内存卡 1 的电池电压降低到低于规定水平。	更换电池。	
1602	在内存卡 2 的电池电压降低到低于规定水平。	更换电池。	QnA

* 1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
2000	UNIT VERIFY ERR.	单元号码	—	关闭/打开	闪烁/打开	停止/继续 *2	当执行 END 指令时	
2100	SP. UNIT LAY ERR.	单元号码	—	关闭	闪烁	停止	当电源打开/复位时	
2101								
2102								
2103								
2104								
2105								
2106								
2107								

*1 在圆括号()里的字符表示存储个别信息的特殊寄存器的号码。

*2 错误出现时的 CPU 操作状态能在参数下设定。(LED 显现会相应地改变)

*1 在圆括号()里的字符表示存储个别信息的特殊寄存器的号码。

	错误代码 (SD0) *1	错误内容和原因	修改方法	相应 CPU
		功能版本为 A 的高性能型 QCPU 装载在多 CPU 系统组态里。	改变功能版本为 A 的高性能型 QCPU 为功能版本为 B 的高性能型 QCPU。	功能版本为 B 或以后的 QCPU
	2000	I/O 模块信息电源 ON 被改变。 ● I/O 模块(或特殊功能模块)没有正确安装或安装在基板上。	<ul style="list-style-type: none"> ● 使用外围设备来读取错误的公共信息,同时检查和改变对应于数字值(模块号码)的模块。 ● 或者,在外围设备监视从 SD1400 到 SD1431 的特殊寄存器,同时改变位有值 1 的输出模块的保险丝。 ● 当显示连接到主基板或扩展基板时,检查扩展电缆的连接状态和连接元件的接地状态。 	○ Rem
		在参数 I/O 分配里,装载 Q160 的插槽设定为智能功能模块或中断模块以外的其他模块。	再次设定使得参数 I/O 分配和实际装载状态相匹配。	功能版本为 B 或以后的 QCPU
	2100	(1) 在参数 I/O 分配设定里,一个智能功能模块被分配到一个为 I/O 模块保留的位置或反之。 (2) 在参数 I/O 分配设定里,CPU 模块之外的模块被分配到一个为 CPU 模块保留的位置,或反之。 (3) 一个通用开关被设定到没有通用开关的模块上。 在参数 I/O 分配设定里,智能功能模块被分配到一个为 I/O 模块保留的位置或反之。	<ul style="list-style-type: none"> ● 复位参数 I/O 分配设定,使其与智能功能模块和 CPU 模块的实际状态相符合。 ● 删除通用开关设定。 复位参数 I/O 分配设定,使其与特殊功能模块的实际状态相符合。	QCPU Rem QnA
	2101	13 个或更多能够初始化中断到 CPU 模块的 A 系列特殊功能模块(除了 A1S161 外)已经安装。	减少启动 CPU 模块的中断的 A 系列功能模块(除了 A1S161 外),使其为 12 个或更少。	QCPU
		13 个或更多能传送到中断到 CPU 模块的的特殊功能模块(不包括 A1S61)已经被安装。	保持能初始化中断(除了 A1S61 外)的特殊功能模块数量为 12 个或更少。	QnA
	2102	7 个或更多 A1SD51S 被安装。	保持 A1SD51S 为 6 或更少。	QCPU
		7 个或更多串行通信模块被安装(排除 ASS61)。	保持安装的串行通信模块(不包括 A(1S)J71QC24)的数量为 6 或更少。	QnA Rem
	2103	(1) 2 个或更多 Q160/A1S161 模块被装载在单 CPU 系统里。 (2) 2 个或更多 Q160/A1S161 模块在多 CPU 系统里被设定给同样的控制 CPU。 (3) 2 个或更多 A1S161 模块被装载在多 CPU 系统里。 2 个或更多 Q160, A1S161 中断模块被安装。 Q160 被装载。 2 个或更多 A1S161 中断模块被安装。	<ul style="list-style-type: none"> ● 在单 CPU 系统里减少装载的 Q160/A1S161 模块的数量为 1。 ● 在多 CPU 系统里改变对于同样控制 CPU 的 Q160/A1S161 模块的数量为 1。 ● 在多 CPU 系统里,减少 A1S161 模块的数量为 1 在多 CPU 系统里当用每个 QCPU 的中断模块时,改为 Q160。(使用一个 A1S161+ 最多 3 个 Q160 模块或只有 Q160 模块)。 只安装 Q160, A61 模块。 去除 Q160。 只安装 A161 模块。	功能版本为 B 或以后的 QCPU QCPU Rem QnA
	2104	在 MELSECNET/H/MINI 自动刷新参数设定里,设定的模块分配不同于连接系统中站点号码上的实际模块类型。	重新设置参数 MELSECNET/H/MINT 自动刷新单元模块分配设定,使得它符合实际链接的模块站号码。	QnA
	2105	有很多特殊功能模块可以使用专用指令分配(安装模块的数字)。(下列指示的总数高于 1344)。 (安装的 AD59 模块的数量 × 5) (安装的 AD57(S1)/AD58 模块的数量 × 8) (安装的 AJ71C24(S3/S6/S8)模块的数量 × 10) (安装的 AJ71UC24 模块的数量 × 10) (安装的 AJ71C21(S1)模块的数量 × 29) (安装的 AJ71PT32-S3/AJ71T32-S3 模块的数量 × 125) * (安装的 AJ71QC24(R2, R4)模块的数量 × 29) (安装的 AJ71ID1(2)-R4 模块的数量 × 8) (安装的 AD75 模块的数量 × 12) 总数 > 1344	减少安装的特殊功能模块的数量。 *: 当使用扩展模式时。	QnA
	2106	<ul style="list-style-type: none"> ● 5 个或更多 MELSECNET/H/H 模块被安装在整个多 CPU 系统里。 ● 5 个或更多 Q 系列以太网接口模块在整个多 CPU 系统里装载。 (1) 5 个或更多 MELSECNE/H 模块被安装。 (2) 5 个或更多 Q 系列以太网接口模块被安装。 (3) 在 MELSECNET/H/10 网络系统里同样的网络号码或站号码存在。 (1) 5 个和更多 AJ71QLP21 模块被安装。 (2) 3 个和更多 AJ71Q12 模块被安装。 (3) 安装的 AJ71Q12 模块总数超过 5。 (4) 在 MELSECNET/H/10 的网络系统里,存在相同的网络号码或类似站号码。 (5) 在 MELSECNET/H 的数据链接系统里同时存在 2 个或更多主站和装站站。	<ul style="list-style-type: none"> ● 在整个多 CPU 系统里,减少 MELSECNET/H/H 模块到 4 或更少。 ● 在整个多 CPU 系统里,减少 Q 系列以太网模块到 4 或更少。 <ul style="list-style-type: none"> ● 减少 MELSECNET/H/H 模块为 4 或更少。 ● 减少 Q 系列以太网模块为 4 或更少。 ● 检查网络号码和站号码。 <ul style="list-style-type: none"> ● 减少 AJ71QLP21 和 AJ71QBR11 模块为 4 或更少。 ● 减少 AJ71AP21/R21 和 AJ71AT21B 模块为 2 个或更少。 ● 减少 AJ71QLP21, AJ71QBR11, AJ71AP21/R21 和 AJ71AT21B 模块总数为 4 或更少。 ● 检查网络号码和站号码。 ● 检查站号码。 	功能版本为 B 或以后的 QCPU QCPU Rem QnA
	2107	在参数 I/O 分配设定里的头 X/Y 设定也是另一模块的头 X/Y。	复位参数 I/O 分配设定,使得与特殊功能模块的实际状态相符合。	○ Rem

*1 在圆括号()里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
2108	SP. UNIT LAY ERR.	单元号码	—	关闭	闪烁	停止	在电源打开/复位时	
2109 *6								
2110	SP. UNIT ERROR	单元号码	程序错误位置	关闭/打开	闪烁/打开	停止/继续 *2	当执行执行指令时	
2111	SP. UNIT ERROR	单元号码	程序错误位置	关闭/打开	闪烁/打开	停止/继续 *2	当执行指令时	
2112		单元号码	程序错误位置	关闭/打开	闪烁/打开	停止/继续 *2	当指令执行从 STOP→RUN 时	
2113		FFFF _H (固定)						
2114		单元号码	程序错误位置	关闭/打开	闪烁/打开	停止/继续 *2	当指令执行时	
2115								
2116								
2117								
2118								
2120	SP. UNIT LAY ERR.	—	—	关闭	闪烁	停止	在电源打开/复位时	
2121								
2122								
2124								
2125								
2126		单元号码	—	关闭	闪烁	停止	在电源打开/复位时	
2150	SP. UNIT VER. ERR.	单元号码	—	关闭	闪烁	停止	在电源打开/复位时	
2200	MISSING PARA.	文件名称/驱动名称	—	关闭	闪烁	停止	在电源打开/复位时	
2210	BOOT ERROR	文件名称/驱动名称	—	关闭	闪烁	停止	在电源打开/复位时	

*1 圆括号 () 中的文字为特殊寄存器编号, 其中存储着单个信息。

*2 可以在参数中设定发生错误时的 CPU 工作状态。(LED 显示器将作相应变化)

*6 只能在备用系统中检测到。

错误代码 (SD0) *1	错误内容和原因	修改方法	相应 CPU
2108	(1) A2USCPU 专用的网络模块 A1SJ71LP21, A1SJ71BR11, A1SJ71AP21, A1SJ71AR21, 或 A1SJ71AT2B 已经被安装。 (2) Q2ADE 专用的网络模块 A1SJ71QLP21 或 A1SJ71QBR11 已经被安装。	改变网络模块为 MELSECNE/H 模块。	QCPU
2109 *6	同 ANUCPU 一起使用的 A(1S)J71LP21 或 A(1S)J71BR11 已经安装。	改变网络模式为 A(1S)J71QLP21 或 A(1S)J71QBR11。	QnA
2110	当冗余系统处于备份模式时, 控制系统和备用系统模块组态不同。	检查备用系统的模块组态。	Q4AR
2111	没有装载的站使用目标是 CPU 共享模块的指令来指定。	使用外围设备读取错误的个别信息, 检查对应于值(程序错误位置)的程序, 同时修改。	功能版本为 B 或以后的 QCPU
2112	(1) 由 FROM/TO 指令集指定的位置不是特殊功能模块。 (2) 访问的特殊功能模块, 网络模块有故障。	● 使用外围设备读取错误的个别信息, 检查对应于值(程序错误位置)的程序, 同时在必要时修改。 ● 访问的特殊功能模块有硬件故障。因此, 更换有故障的模块。也可以联系最近的三菱代理商。	○
2113	通过链接直接元件(J□\□)指定的位置不是网络模块。		
2114	(1) 特殊功能模块以外的模块通过特殊功能模块专用指令来指定。或者, 它不是相应的特殊功能模块 (2) 通过特殊功能模块专用指令指定的模块类型和通过参数 I/O 分配指定的模块有区别 例如)尽管实际使用 AJ71Q24N, 也设定 AJ71QC24。	(1) 使用外围设备读取错误的个别信息, 检查对应于值(程序错误部分)的特殊功能模块专用指令(网络指令)来进行改进。 (2) 使用和特殊功能模块专用指令设定相对应的参数 I/O 分配, 设置模块类型。	QnA Q4AR
2115	(1) 智能功能模块以外的模块通过特殊功能模块专用指令来指定。或者, 它不是相对应的特殊功能模块。 (2) 没有通过网络专用指令指定的网络号码。或继电器目标网络不存在。 (3) 模型被专用指令错误登记。	使用外围设备读取错误的个别信息, 检查对应于值(程序错误部分)的特殊功能模块专用指令(网络指令)来进行改进。	QCPU Rem
2116	除了网络模块外的模块被网络专用指令指定。		○
2117	一个指令, 在执行时指定其它站, 被用来指定主机 CPU。(不允许主机 CPU 被指定的指令)		
2118	一个指令, 在执行时指定主机站, 被用来指定其它 CPU(不允许其它 CPU 被指定的指令)	使用外围设备读取错误的个别信息, 检查对应于值的程序(程序错误位置), 同时修改。	功能版本为 B 或以后的 QCPU
2119	(1) 在其他 CPU 控制下不允许被指定的指令被用做相似的任务。 (2) 对于其他 CPU 控制的模块 A 或 QNA 指令被执行。		
2120	专用于多 CPU 系统的指令被指定时, CPU 模块不能被指定。		
2121	当在多 PLC 系统里的过程 CPU 的参数在线模块改变设定已经设定为使能时, 被其他 CPU 控制的智能功能模块已经在 FROM 指令/智能功能模块元件(L□\G□)中被指定。	● 当过程 CPU 的在线模块改变在多 PLC 系统里执行时, 修改程序使得不能访问到被另一 PLC 控制的模块。 ● 当访问被设定到多 PLC 系统里被另一 PLC 控制的模块时, 在过程 CPU 的参数中设定在线模块变化设置为禁止。	○ 序列号 4122 个或之后
2122	Q□B 和 QA1S□B 的位置不正确。	检查基板的位置。	
2123	CPU 模块安装在 CPU 插槽之外的插槽上, 或插槽 0 到 2 之上。	检查 CPU 模块的安装位置同时将它重新安装在正确插槽。	
2124	QA1S□B 用于主基板。	用 Q3□B 作为主基板。	
2125	(1) 模块被安装在第 65 或更高的插槽。 (2) 模块安装在和基板分配设定一起指定的插槽数之后的插槽内。 (3) 模块安装在 4096 点之后的 I/O 点。 (4) 安装在 4096 点的模块占据较高点。	● 去除安装在 65 号插槽和其后的模块。 ● 去除安装在和基板分配设定一起指定的插槽数之后插槽中的模块。 ● 去除安装在 4096 点之后的 I/O 点的模块。 ● 改变最后的模块为不超过 4096 点的模块。	QCPU Rem
2126	(1) QCPU 不能辨认的模块已经安装。 (2) 智能功能模块没有响应。	● 安装一个可使用的模块 ● 智能功能模块有硬件故障。因此, 更改有故障的模块。或者, 联系最近的三菱代理商。	
2127	在多 CPU 系统的 CPU 模块位置属于下列一个。 (1) 在 QCPU 和 QCPU/运动控制器之间有空置的插槽。 (2) 除了高性能型 QCPU/过程 CPU (包括运动控制器)的模块安装在高性能型 QCPU/过程 CPU 的左侧。	● 清除在 CPU 模块中的空置插槽(在 CPU 模块的右侧设置空闲插槽)。 ● 卸载安装在高性能型 QCPU/过程 CPU 中的 QCPU 之外的模块, 并用高性能型 QCPU/过程 CPU 代替它们。 在 QCPU 的右侧安装运动控制器。	功能版本为 B 或以后的 QCPU
2150	在多 CPU 系统, 与多 CPU 系统不兼容的智能功能模块的控制 CPU 设定为 1 号 CPU 之外的 CPU。	● 改变智能功能模块为与多 CPU 系统兼容的模块(功能版本为 B)。 ● 改变与多 CPU 系统不兼容的智能功能模块的控制 CPU 的设定为 1 号 CPU。	
2200	在由 DIP 开关指定的作为有效驱动的驱动器上, 没有参数文件。	● 检查和修改使能驱动开关的参数的设置。 ● 在参数使能驱动开关指定的文件上放置参数文件。	○
2210	启动文件的内容不正确	检查启动设置。	QCPU
2211	尽管启动 DIP 开关打开, 在参数使能驱动开关指定的驱动上没有启动文件。	● 检查并修改参数使能驱动开关的设置。 ● 在参数使能驱动开关指定的驱动上放置启动文件。	QnA

*1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

*6 只有在冗余系统的备用系统能被检测。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
2300	ICM. OPE. ERROR	文件名称/驱动名称	—	关闭/打开	闪烁/打开	停止/继续 *2	当内存卡插入或去除时	
2301	ICM. OPE. ERROR	文件名称/驱动名称	—	关闭/打开	闪烁/打开	停止/继续 *2	当内存卡插入或去除时	
2302								
2400	FILE SET ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	电源打开/复位	
2401	FILE SET ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	电源打开/复位	
2402								
2410	FILE OPE. ERROR	文件名称/驱动名称	程序错误位置	关闭/打开	闪烁/打开	停止/继续 *2	指令执行时	
2411								
2412								
2413								
2500	CAN' T EXE. PRG.	文件名称/驱动名称	—	关闭	闪烁	停止	电源打开/复位	
2501								
2502								
2503								
2504								
3000	PARAMETER ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	电源打开/复位 STOP → RUN	
3001								
3002								
3003	PARAMETER ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	当 END 指令执行时	
	PARAMETER ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	电源打开/复位 STOP → RUN	

*1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

*2 错误出现时的 CPU 的操作状态能在参数下设定。(LED 显现会相应地改变)

错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
2300	(1) 内存卡在没有将内存卡进入/出去开关切换为 OFF 的情况下被去除。 (2) 尽管内存卡没有实际安装, 但内存卡输入/输出开关打开。	<ul style="list-style-type: none"> 在把内存卡输入/输出开关置为 OFF 后, 去除内存卡。 在插入内存卡后, 打开内存卡插入开关。 	○
2301	(1) 内存卡没有格式化。 (2) 内存卡格式化状态不正确。	<ul style="list-style-type: none"> 格式化内存卡。 重新格式化内存卡。 	○
2302	安装了不能与 CPU 模块一起使用的内存卡。	检查内存卡。	
2400	自动写入到标准 ROM 在与自动写入到标准 ROM 不兼容的 CPU 模块上执行。(在源文件中被选取的自动写入到标准 ROM 的内存卡装备好同时参数可能驱动谁多能工为内存卡)。	<ul style="list-style-type: none"> 在与自动写入到标准 ROM 兼容的 CPU 模块上执行自动写入到标准 ROM。 使用 GX DEVELOPER, 将参数和程序写入到标准 ROM。 对于自动写入到标准 ROM 没有设定的需要更换内存卡, 同时执行从内存卡的启动操作。 	功能版本为 B 或以后的 QCPU
	在参数中的 PLC 文件设定里指定的文件不能找到。	使用外围设备读取错误的个别信息, 检查对应于数字值的参数驱动名称和文件名称正确。 使用参数建立文件, 同时安装到 CPU 模块里。	○
	添加到带功能版本为 B 的 QNACPU 的以太网参数, 被设定给没有带功能版本为 B 的 QNACPU。	用功能版本为 B 的来替换 QNACPU。 删除以太网参数。	QnA
2401	执行启动操作或自动写入到标准 ROM 程序, 超出了内存容量。	<ul style="list-style-type: none"> 检查和修改参数(启动设定)。 删除在程序内存里不必要的文件。 在参数中选择用于启动的“清除程序内存”, 这样在内存清除后启动开始。 	功能版本为 B 或以后的 QCPU
	在参数 PLC RAS 设定故障历史区域指定的文件不能创建。	使用外围设备读取错误的个别信息, 检查对应于数字值的参数驱动名称和文件名称, 并改正错误。 检查在内存卡里的剩余空间。	○
	2402 尽管文件寄存器在成对设定和/跟踪设定里已经设置, 文件寄存器不存在。	确认文件寄存器和参数。	Q4AR
2410	指定的程序在程序内存里不存在。 这在执行 ECALL, EFCALL, PSTOP, PSCAN, POFF 或 PLOW 等指令时, 该错误可能出现。	通过外部设备读取单个错误信息, 检查并确保程序和数值(程序位置)相一致, 改正错误。 利用参数创建文件, 将其加载至 CPU 模块。	○
2411	指定程序位于程序存储器中, 但是没有在参数对话框的程序设定中登记。当执行 ECALL, EFCALL, PSTOP, PSCAN, POFF 或 PLOW 指令时, 可能出现此错误。	使用外围设备读取错误的个别信息, 检查并确保程序和数值(程序位置)相一致, 改正错误。	
2412	SFC 程序文件是不能通过顺序程序指定的。		
2413	没有数据写入到顺序程序指定的文件中。	使用外围设备读取错误的个别信息, 检查并确保程序和数值(程序位置)相一致, 改正错误。 检查并确保指定文件没有写保护。	
2500	(1) 一个程序文件使用的元件超出在 PLC 参数元件设置中的设置范围。 (2) 在 PLC 参数设定改变后, 只有参数写入 PLC。	(1) 使用外围设备读取错误的个别信息, 检查从而确信对应于数字值(文件名称)的参数元件分配设置和程序文件元件分配, 如果有必要可以修改。 (2) 如果 PLC 参数元件设置变化, 成批写入参数和程序文件到 PLC。	○
	2501 尽管没有设置在参数程序设置中, 仍然有多个程序文件。	编辑参数程序设定为“是”。 或者, 删除不必要的程序。	○
	2502 程序文件不正确。 或者, 文件内容不是顺序程序的那些内容。	检查是否程序版本是***.QPG, 检查文件内容确信它们用于顺序文件。	
	2503 根本没有程序文件。	● 检查程序组态。	
	2504 2 个和更多 SFC 常规程序或控制程序已经被指派。	● 检查参数和程序组态。	
3000	在多 CPU 系统里, 在其他 CPU 控制下的智能功能模块在参数的中断指针设置中被指定。	<ul style="list-style-type: none"> 在主机 CPU 控制下, 指定智能功能模块的起始 I/O 号码。 删除参数的中断指针设置。 	功能版本为 B 或以后的 QCPU
	对于定时器时间限制设置的参数设置, RUN-PAUSE 触点, 公共指针号码, 一般数据处理, 空置插槽的数量, 和系统中断设置在能被 CPU 模块使用的范围之外。	<ul style="list-style-type: none"> 使用外围设备读取错误的详细信息, 检查对应于数字值的参数条目, 同时在必要时修改。 在修改参数设定后错误依然存在, 可能的原因是 CPU 模块的内置 RAM 或程序内存, 或内存卡的内存错误。因此, 更换有故障的模块。也可以联系最近的三菱代理商。 	○ Rem
	3001 参数内容被破坏。		○
	3002 在 PLC 文件设置参数里当选择“使用下列文件”作为文件寄存器时, 尽管文件寄存器容量已经设置指定文件并不存在。		
	3003 多 CPU 系统的自动刷新范围超过文件寄存器容量。	在整个范围内改变文件寄存器为一个刷新使能。	功能版本为 B 或以后的 QCPU
	在参数元件设置中设定的元件数量超出了可能的 CPU 模块范围。	<ul style="list-style-type: none"> 使用外围设备读取错误的详细信息, 检查对应于数字值(参数号码)的参数条目, 同时在必要时修改。 在修改参数设定后错误依然存在, 可能的原因是 CPU 模块的内置 RAM 或程序内存, 或内存卡的内存错误。因此, 更换有故障的模块。也可以联系最近的三菱代理商。 	○

* 1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
3004	PARAMETER ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	在电源打开/复位时 STOP → RUN	
3006	PARAMETER ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	在电源打开/复位时 STOP → RUN	
3009	PARAMETER ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	在电源打开/复位时 STOP → RUN	
3010								
3012								
3013								
3100	LINK PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	在电源打开/复位时 STOP → RUN	
3101								
3102	LINK PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	在电源打开/复位时 STOP → RUN/当 END 指令执行时。	
3103	LINK PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	在电源打开/复位时 STOP → RUN	

*1 在圆括号()里的字符表示存储个别信息的特殊寄存器的号码。

	错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
	3004	参数文件不正确。 或者文件内容与参数无关。	检查是否参数文件版本是否是 * * *.QPA, 同时检查文件内容确认为参数。	
	3006	(1) 高速中断参数在 Q02CPU 设置。 (2) 高速参数在多 CPU 系统中设置。 (3) 当 QA1S□B 或 QA□B 使用时, 高速中断参数被设定。 (4) 没有模块安装在通过高速中断参数指定的 I/O 地址。	<ul style="list-style-type: none"> ● 删除 Q02CPU 的高速中断参数设置, 为了使用高速中断, 改变 CPU 模块为 Q02HCPU 之一。 ● 为了使用多 CPU 系统, 删除高速中断参数的设置。为了使用高速中断, 改变系统为单 CPU 系统。 ● 使用任一 QA 或 QB, 删除高速中断参数的设置, 为了使用高速中断, 不要使用 QA1 或 QB。 ● 通过高速中断参数重新检查指定的 I/O 地址。 	功能版本为 B 或以后的 QCPU
	3009	在多 CPU 系统里, AnS, A, Q2AS 和 QnA 的模块已经设定为多控制 CPU。	在一个 CPU 下重新设置参数 I/O 分配来控制它们。(在多 CPU 系统改变所有 CPU 的参数)。	功能版本为 B 或以后的 QCPU
	3010	CPU 模块的参数设置号码不同于在多 CPU 系统里的实际号码。	匹配(多 CPU 设置的预设数—在 I/O 分配中的 CPU(空置))同实际装载的 CPU 的数量。	
	3012	多 CPU 设置或控制 CPU 设置不同于在多 CPU 系统里的参考 CPU 的设置。	匹配在参数里的多 CPU 设置或控制 CPU 设置同参照 CPU 的相应部分(1 号 CPU)。	
	3013	多 CPU 自动刷新设置是多 CPU 系统中的如下任一设置。 (1) 当一个位元件指定为刷新元件时, 16 的倍数以外的数字被指定为刷新启动元件。 (2) 指定的元件不是可能被指定的元件。 (3) 传送点数是奇数。	在多 CPU 自动刷新参数中检查下列情况, 并进行修改。 <ul style="list-style-type: none"> ● 当指定一个位元件时, 刷新启动元件指定为 16 的倍数。 ● 指定可以被指定为刷新元件的元件。 ● 设定传送点数为偶数。 	
	3100	在多 CPU 系统里受控于其他 CPU 的 MELSECNET/H/H 在 MELSECNET/H/H 的网络设定参数里被指定为起始 I/O 数字。 在普通站中操作的 MELSECNET/H/H 的网络参数被重新写入控制站, 或者在控制站操作的 MELSECNET/H/H 的网络参数被重新写入普通站。	<ul style="list-style-type: none"> ● 删除受控制于其他 CPU 的 MELSECNET/H 的网络参数。 ● 改变受控于主 CPU 的 MELSECNET/H 的起始 I/O 号码的设定。 复位 CPU 模块。	功能版本为 B 或以后的 QCPU
		(1) 实际安装的模块号码不同于 MELSECNET/H/H 里模块设置参数中指定的号码。 (2) 实际安装模块的起始 I/O 号码不同于指定的 MELSECNET/H/H 里的网络参数。 (3) 参数中的一些数据不能处理。 (4) 当电源打开时, MELSECNET/H 的站类型被改变。(改变站类型需要 RESET → RUN)。	<ul style="list-style-type: none"> ● 检查网络参数和安装状态, 如果有区别, 匹配网络参数和安装状态, 如果任何网络参数被修改, 写入到 CPU 模块。 ● 确认扩展基板的扩展段数量的设置。 ● 检查扩展基板和连接器的连接状态, 当显示元件连接到主基板和扩展基板时, 检查连接状态。 ● 如果以上操作后仍有错误出现, 联系最近的三菱代理商。 	QCPU
		尽管 QNACPU 是控制站或主站, 但网络参数没有写入。	<ul style="list-style-type: none"> ● 修改和写入网络参数。 ● 在修改后仍出现错误, 表示是硬件故障。因此, 更换有故障的硬件。也可以联系最近的三菱代理商。 	QnA
		连接刷新范围超过了文件寄存器的容量。	更改使能全部范围刷新的文件寄存器的文件。	功能版本为 B 或以后的 QCPU
	3101	(1) 当 MELSECNET/H 模块的站号码为 0 时, 内部 PLC 网络参数设定已经建立 (2) 当 MELSECNET/H 模块的站号码不为 0 时, 远程主参数设置已经建立。	在参数里修改 MELSECNET/H 模块的类型和站号码从而符合使用过的系统。	
		(1) 通过参数指定的网络号码不同于实际安装的网络的号码。 (2) 通过参数指定的起始 I/O 号码不同于实际安装的 I/O 单元的号码。 (3) 通过参数指定的网络等级不同于实际安装的网络等级。 (4) MELSECNET/H, MELSECNET/10 的网络刷新参数, 超出了指定区域。	<ul style="list-style-type: none"> ● 检查网络参数和安装状态, 如果有区别, 匹配网络参数安装状态, 如果任何网络参数被修改, 写入到 CPU 模块。 ● 确认扩展基板的扩展级数量的设置。 ● 检查扩展基板和连接器的连接状态, 当显示元件连接到主基板和扩展基板时, 检查连接状态。 ● 在进行如上操作后仍有错误出现, 联系最近的三菱代理商。 	○
	3102	网络参数检查在网络模块中进行时, 发现错误。 指定到 MELSECNET/H 和 MELSECNET/10 的参数不正常。	<ul style="list-style-type: none"> ● 修改和写入网络参数。 ● 在修改后仍出现错误, 表示是硬件故障。因此, 更换有故障的硬件。也可以联系最近的三菱代理商。 	
	3103	在多 CPU 系统, 在其他站控制下的 Q 系列以太网接口模块指定作为以太网设定参数的起始 I/O 号码。	<ul style="list-style-type: none"> ● 删除受其他站控制的 Q 系列以太网接口模块的以太网设置参数。 ● 改变受控于主站的 Q 系列以太网接口模块的起始 I/O 号码设定。 	功能版本为 B 或以后的 QCPU
		(1) 尽管以太网模块的数量在参数里设定为 1 或更多, 实际的单元安装数量是 0。 (2) 以太网模块设置参数的起始 I/O 号码不同于实际安装模块的号码。	<ul style="list-style-type: none"> ● 修改和写入网络参数。 ● 在修改后仍出现错误, 表示是硬件故障。因此, 更换有故障的硬件。也可以联系最近的三菱代理商。 	○ Rem
		(1) 在参数设定的 I/O 号码的位置不存在 AJ71QE71。 (2) I/O 号码指定重叠。 (3) 参数数量和装载的 AJ71QE71 不同。 (4) 以太网(参数+专用指令)设定大于 5。		QnA

* 1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
3104	LINK PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	当电源打开/复位 S1OP→RUN	
3105								
3106	LINK PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	执行 END 指令	
	LINK PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	当电源打开/复位 S1OP→RUN	
3107	LINK PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	当电源打开/复位 S1OP→RUN	
3200	SFC PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	STOP→RUN	
3201								
3202								
3203								
3300	SP. PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	当电源打开/复位 S1OP→RUN	
3301							执行 END 指令	
							当电源打开/复位 S1OP→RUN	
3302							SP. PARA. ERROR	
3303								
3400	REMOTE PASS. ERROR	—	—	关闭	闪烁	停止	当电源打开/复位 S1OP→RUN	
3401								
4000	INSTRCT CODE ERR.	程序错误位置	—	关闭	闪烁	停止	当电源打开/复位 S1OP→RUN	
4001								
4002								
4003								
4004								
4010	MISSING END INS.	程序错误位置	—	关闭	闪烁	停止	当电源打开/复位 S1OP→RUN	
4020	CAN' T SET (P)	程序错误位置	—	关闭	闪烁	停止		
4021								
4030	CAN' T SET (I)	程序错误位置	—	关闭	闪烁	停止		

*1 在圆括号()里的字符表示存储个别信息的特殊寄存器的号码。

*2 错误出现时的 CPU 的操作状态能在参数下设定。(LED 显现会相应地改变)

错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
3104	(1) 以太网和 MELSECNET/10 使用同样的网络号码。 (2) 通过参数设置的网络号码, 站号码或组号码超出范围。 (3) I/O 号码被指定给所使用的 CPU 模块。 (4) 以太网专用参数设置不正常。	<ul style="list-style-type: none"> ● 修改和写入网络参数。 ● 在修改后仍出现错误, 表示是硬件故障。因此, 更换有故障的硬件。也可以联系最近的三菱代理商。 	○ Rem
3105	在多 CPU 系统, 受其他站控制的 Q 系列 CC-Link 模块被指定为 CC-Link 设定参数的起始 I/O 号码。	<ul style="list-style-type: none"> ● 删除受其他站控制的 Q 系列 CC-Link 模块的 CC-Link 设置参数。 ● 改变受控于主机站的 Q 系列 CC-Link 模块的起始 I/O 号码的设置。 	功能版本为 B 或以后的 QCPU
	(1) 尽管在网络参数里设定里的 CC-Link 模块的数量是 1 或更多, 实际安装单元的数量是 0。 (2) 普通参数中的起始 I/O 号码不同于实际安装模块的数量。 (3) CC-Link 模块数量设置参数的站点等级不同于实际安装站点的等级。	<ul style="list-style-type: none"> ● 在修改网络参数后写入。 ● 在修改后仍出现错误, 表示是硬件故障。因此, 更换有故障的硬件。也可以联系最近的三菱代理商。 	○ Rem
	以太网参数的内容不正确。	在修改参数后写入。	QnA
3106	CC-Link 刷新范围超过了文件寄存器容量。	在整个范围内, 改变用于刷新使能的文件寄存器文件。	功能版本为 B 或以后的 QCPU
	用于 CC-Link 的网络刷新参数超出范围。	检查参数设定。	QCPU Rem
3107	(1) CC-Link 参数设定不正确。 (2) 安装的 CC-Link 模块的版本不允许设定模式。	检查参数设定。	○ Rem
3200	参数设定非法 (1) 尽管在 PLC 参数对话框的 SFC 设定中块 0 被设定为“自动启动”, 但块 0 不存在。	使用外围设备读取错误的公共信息, 检查对应于数字值(程序错误位置)的错误步, 并改正问题。	○
3201	块参数设定为非法。		
3202	在参数里指定的步继电的数字少于被程序使用的数字。		
3203	SFC 程序在参数里设定的执行类型不是扫描执行类型。		
3300	在 GX 组态器上设定的智能功能模块参数中的起始 I/O 号码不同于实际 I/O 号码。	检查参数设置。	QCPU Rem
3301	智能功能模块的刷新设置超出了文件寄存器容量。	在整个范围内改变用于允许刷新的智能功能模块的文件寄存器文件。	功能版本为 B 或以后的 QCPU
	智能功能模块的刷新参数设置超出了现有范围。	检查参数设定	QCPU Rem
3302	智能功能模块的刷新参数不正常。	检查参数设定。	QCPU
3303	在多 CPU 的系统里, 自动刷新设置或类似的参数设定被设定到其他站的控制下的智能功能模块里。	<ul style="list-style-type: none"> ● 删除受控于其他 CPU 的智能功能模块的自动刷新设置或类似的参数设置。 ● 改变受控于主机 CPU 的智能功能模块的自动刷新或类似参数的设置。 	功能版本为 B 或以后的 QCPU
3400	在远程口令文件里的目标模块的起始 I/O 号码的设定不在 0H 到 0FF0H 之间。	改变目标模块的起始 I/O 号码到 0H 到 0FF0H 范围内。	
3401	由于下列的某个原因, 被指定作为远程口令文件的起始 I/O 号码的位置不正确。 (1) 模块没有安装。 (2) 不是 Q 兼容智能模块(I/O, A, QnA 模块)。 (3) Q 系列串行通信模块, 调制解调器接口模块或以太网模块之外的智能功能模块。 (4) 功能版本为 A 的 Q 系列串行通信模块或以太网模块。	在远程口令文件的起始 I/O 号码指定的位置安装功能版本为 B 的 Q 系列串行通信模块, 调制解调器接口模块或以太网模块。	
	被其他 CPU 控制的功能版本为 B 的 Q 系列串行通信模块, 调制解调器接口模块或以太网模块在多 CPU 系统中被指定。	(1) 对于连接到主机 CPU 的功能版本为 B 的以太网模块, 改变它。 (2) 删除远程口令设定。	
4000	(1) 程序包含一个不能被解码的指令号码。 (2) 一个不能使用的指令包括在程序里。	使用外围设备读取错误的公共信息, 检查对应于数字值(程序错误位置)的错误步, 同时改正问题。	○
4001	尽管它不是一个 SFC 程序, 但程序包含一个 SFC 专用指令。		○
4002	(1) 程序指定的扩展指令有一个错误的指令名称。 (2) 通过程序指定的扩展指令不能通过指定模块来执行。		○ Rem
4003	程序指定的扩展指令的元件数量不正确。		○
4004	程序(元件)指定的扩展指令不能使用。		○
4010	程序中没有 END(FEND)指令。		○
4020	被程序使用的整个内部文件点的数量超过了在参数里设定的内部文件点的数量。		○
4021	由个别文件使用的公共指针号码重叠。		○
4030	由个别文件使用的位置分配指针重叠。		○

* 1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
4100	OPERATION ERROR	程序错误位置	—	关闭	闪烁	停止/继续 *2	当指令执行时	
4101								
4102								
4103								
4104								
4107								
4108								
4109								
4200	FOR NEXT ERROR	程序错误位置	—	关闭	闪烁	停止	当指令执行时	
4201	FOR NEXT ERROR	程序错误位置	—	关闭	闪烁	停止	当指令执行时	
4202	FOR NEXT ERROR	程序错误位置	—	关闭	闪烁	停止	当指令执行时	
4203								
4210	CAN' T EXECUTE (P)	程序错误位置	—	关闭	闪烁	停止	当指令执行时	
4211								
4212								
4213								
4220	CAN' T EXECUTE (I)	程序错误位置	—	关闭	闪烁	停止	当指令执行时	
4221								
4223								
4230	INST. FORMAT ERR.	程序错误位置	—	关闭	闪烁	停止	当指令执行时	
4231								
4235								
4300	EXTEND INST. ERR.	程序错误位置	—	关闭/打开	闪烁/打开	停止/继续 *2	当指令执行时	
4301								
4400	SFCP. CODE ERROR	程序错误位置	—	关闭	闪烁	停止	STOP → RUN	
4410	CAN' T SET (BL)	程序错误位置	—	关闭	闪烁	停止	STOP → RUN	
4411								
4420	CAN' T SET (S)	程序错误位置	—	关闭	闪烁	停止	STOP → RUN	
4421								
4422								

*1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

*2 错误出现时的 CPU 的操作状态能在参数下设定。(LED 显现会相应地改变)

	错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
	4100	指令不能处理包含数据。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	○
	4101	(1) 用于指令处理的被指定元件号码超出了使用范围。 (2) 或者，通过指令指定的元件存储的数据和常量超过使用范围。		
	4102	在多 CPU 系统，链接直接元件(J□\G□)被指定为其他站控制下的网络模块。	(1) 从程序里删除指定受其他 CPU 控制的网络模块的链接直接元件。 (2) 使用链接直接元件，指定受控于主机 CPU 的网络模块。	功能版本为 B 或以后的 QCPU
		用专用网络指令指定的网络号码和站号码不正确。 链接直接元件(J□\W□)没有被正确设定。 ● 通过扩展指令指定的模块号码/网络号码/字符串计数超过设定范围。 ● 扩展指令指定的字符串(“”)不能使用。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	○ Rem
	4103	PID 专用指令的配置不正确。		○
	4104	设置的数量超出范围。	使用外围设备读取错误的公共信息，检查和修改对应于数值(程序错误位置)的程序。	Q4AR
	4107	33 个或更多的多 CPU 专用指令从一个 CPU 模块处理。	使用多 CPU 专用指令完成位，提供互锁来防止一个 CPU 模块执行 32 个或更多 CPU 专用指令。	功能版本为 B 或以后的 QCPU
		对于 CC-Link 指令执行次数超过 32。	设定 CC-Link 指令的执行次数为 32 个或更少。	QnA
	4108	当 CC-Link 指令执行时，CC-Link 参数没有设定。	在设定 CC-Link 参数后，执行 CC-Link 指令。	
	4109	用高速中断设置执行 PR/PRC, UDCNT1, UDCNT2, PLSY, PWM, SPD, PLOADP, PUNLOADP, PAWPP 等指令。	删除高速中断参数设置， 当使用高速中断，删除 PR, PRC, UDCNT1, UDCNT2, PLSY, PWM, SPD, PLOAD, PUNLOAD 和 PSWAP 等指令。	功能版本为 B 或以后的 QCPU
	4200	在 FOR 指令执行后，没有 NEXT 执行或者，NEXT 指令数少于 FOR 指令。 尽管没有 FOR 指令被执行，NEXT 仍然执行。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	
	4201	尽管未执行 FOR 指令，仍执行 NEXT 指令。 或者，NEXT 指令数多于 FOR 指令。		
	4202	多于 16 级嵌套被处理。	保持嵌套级在 16 或更低。	
	4203	尽管先前没有 FOR 指令的执行，BREAK 指令被执行。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	
	4210	CALL 指令执行时，在指定的指针处没有子函数。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	○
	4211	在执行子函数程序里，没有 RET 指令。		
	4212	在主程序中，RET 指令在 FEND 指令之前。	保持嵌套级在 16 或更低。	
	4213	多于 16 级嵌套被处理。		
	4220	尽管中断输入出现，但相应的中断指针不存在。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	
4221	被执行的中断程序里 RET 指令不存在。			
4223	在主程序的 FEND 指令前有 IRET 指令。			
4230	CHK 和 CHKEND 指令的数量不相等。			
4231	IX 和 IXEND 指令的数量不相等。			
	4235	CHK 指令的检查条件组态不正确。 或者，在一个低速执行类型程序里，CHK 指令被使用。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	QnA
	4300	MELSECNET/H/MINI-S3 主模块控制指令的指定有错。		
	4301	AD57/AD58 控制指令组态错误。		
	4400	在 SFC 程序里没有 SFCP 或 SFCPEND 指令。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	○
	4410	SFC 程序指定的块号码超出了范围。		
	4411	块号码指定在 SFC 程序里重叠。	减少总的步数在最大值之内。	
	4420	在 SFC 程序里指定的步数超过了 511。		
	4421	在所有 SFC 程序里的总步数超过范围。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	
	4422	步号码指定在 SFC 程序里重叠。		

*1 在圆括号()里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
4500	SFCP. FORMAT ERR.	程序错误位置	—	关闭	闪烁	停止	STOP → RUN	
4501								
4502								
4503								
4504								
4600	SFCP. OPE. ERROR	程序错误位置	—	关闭/打开	闪烁/打开	停止/继续 *2	当指令执行时	
4601								
4602								
4610	SFCP. EXE. ERROR	程序错误位置	—	打开	打开	继续	STOP → RUN	
4611								
4620	BLOCK EXE. ERROR	程序错误位置	—	关闭	闪烁	停止	当指令执行时	
4621								
4630	STEP EXE. ERROR	程序错误位置	—	关闭	闪烁	停止	当指令执行时	
4631								
4632								
4633								
5000	WDT ERROR	时间(设定值)	时间(实际测量值)	关闭	闪烁	停止	一直	
5001								
5010	PRG. TIME OVER	时间(设定值)	时间(实际测量值)	打开	打开	继续	一直	
5011								
6000	PRG. VERIFY ERR. *5	文件名称	—	关闭	闪烁	停止	一直	
6010	MODE VERIFY ERR. *5	—	—	打开	打开	继续	一直	
6100	TRK. MEMORY ERR. *3	—	—	打开	打开	继续	电源打开/复位 /STOP → RUN	
6101							当 END 指令执行时	
6200	CONTROL EXE. *4	转换原因	—	打开	关闭	继续	一直	
6210	CONTROL WAIT. *5	转换原因	—	打开	关闭	继续	一直	
6220	CAN'T EXE CHANGE *4	转换原因	—	打开	打开	继续	转换要求	
6221								
6222								
6230	DUAL SYS ERROR	—	—	打开	打开	继续	一直	

*1 在圆括号()里的字符表示存储个别信息的特殊寄存器的号码。

*3 只能在冗余系统里被检测。能在控制系统或备用系统里被检测。

*4 只能在冗余系统的控制系统里被检测。

*5 只能在冗余系统的备用系统里被检测。

错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
4500	在 SFC 程序里 BLOCK 和 BEND 指令的数量不相等。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	○
4501	在 SFC 程序里从 STEP * 到 TRAN * 到 TSET 到 SEND 的组态不正确。		
4502	在 SFC 程序块里没有 STEP * 指令。		
4503	在 SFC 程序里通过 TSET 指令指定的步不存在。		
4504	在 SFC 程序里通过 TAND 指令指定的步不存在。		
4600	SFC 程序里包含不能处理的数据。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	○
4601	超出能被 SFC 程序指定的元件范围。		
4602	在 SFC 程序里的 START 指令通过 END 指令来处理。		
4610	在 SFC 程序的假定开始处的活动步信息不正确。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。 程序自动受制于初始化启动。	○
4611	当为 SFC 程序指定假定开始时，钥匙开关在 RUN 时被复位。		
4620	启动在 SFC 程序里已经启动的块上执行。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	○
4621	启动在 SFC 程序里不存在的块上尝试。	复位 CPU 模块并再运行，如果出现相同的问题，智能功能模块，CPU 模块或基板有故障，因此，更换有故障的模块或单元。或者联系最近的三菱代理商。	
4630	启动在 SFC 程序里已经启动的块上执行。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	○
4631	启动在 SFC 程序里不存在的块上尝试。	复位 CPU 模块并再运行，如果出现相同的问题，智能功能模块，CPU 模块或基板有故障，因此，更换有故障的模块或单元。或者联系最近的三菱代理商。	
4632	能被 SFC 程序指定的块上有太多同时活动步。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	
4633	在被指定的块上有太多同时活动步。	使用外围设备读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	
5000	对于初始执行类型程序的程序扫描时间超过了在 PLC RAS 参数里设定的初始执行 WDT 时间设置。	使用外围设备读取错误的个别信息，检查数字值，同时如果必要，缩短扫描时间。	○
5001	程序扫描时间超过在 PLC RAS 参数中设定的 WDT 值。		
5010	(1) 程序扫描时间超过在 PLC RAS 设置参数指定的常量扫描设置时间。 (2) 在 PC RAS 设置参数指定的低速程序执行时间超过恒定周期的多余时间。	<ul style="list-style-type: none"> ● 检查恒定扫描设置时间。 ● 检查在参数里的恒定扫描时间和低速程序执行时间，使得恒定扫描的多余时间能被完全保留。 	○
5011	低速执行类型程序扫描时间超过在参数 PLC RAS 设置里的低速执行 WDT 设置。	使用外围设备读取错误的个别信息，检查数字值，同时如果必要，缩短扫描时间。	
6000	在冗余系统的控制系统和备用系统里没有同样的程序和参数。	使得控制系统和备用系统的程序和参数同步。	Q4AR
6010	在冗余系统的控制系统和备用系统里的操作状态不一致。	使得控制系统和备用系统的操作状态同步。	
6100	在初始化时，一个 CPU 模块跟踪内存错误被检测。	这是一个 CPU 模块硬件故障。因此，更换有故障的模块。或者，联系最近的三菱代理商。 当替换 CPU 模块时，首先替换备用系统 CPU，然后是控制系统 CPU。	Q4AR
6101	CPU 模块在跟踪的握手过程中检测到一个错误。	检查其它站的条件。	
6200	冗余系统的备用系统切换为控制系统。	检查控制系统条件。	
6210	冗余系统的控制系统切换为备用系统。	检查控制系统条件。	Q4AR
6220	因为一个错误状态或其它原因，冗余系统的备用系统不能从控制系统切换到备用系统。	检查备用系统条件。	
6221	因为总线转换模式错误，切换不能执行。	这是总线转换模块硬件故障。因此，更换有故障的硬件。或者，联系最近的三菱代理商。	
6222	因为在初始化时，远程 I/O 网络的多任务主站安装在备用站中，切换无效。	检查远程 I/O 网络设置。	
6230	安装在备用类型 CPU 的链接模块是主站。	检查系统组态状态。	

* 1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
7000	MULT CPU DOWN	单元号码	—	关闭	闪烁	停止	一直	
7002							电源打开/复位	
7003							电源打开/复位	
7010	MULTI EXE. ERROR	单元号码	—	关闭	闪烁	停止	电源打开/复位	
7020	MULTI CPU ERROR	单元/模块号码	—	打开	打开	继续	一直	
9000	F**** *6	程序错误位置	警报器号码	打开	关闭	继续	当执行指令时	
9010	<CHK> ERR ***-*** *7	程序错误位置	过失号	打开	关闭			
				USER LED 开	USER LED 开			
9020	BOOT OK	—	—	关闭	闪烁	停止	电源打开/复位	
10000	CONT. UNIT ERROR	—	—	关闭	闪烁	继续	一直	

*1 在圆括号()里的字符表示存储个别信息的特殊寄存器的号码。

*6 ****表示检测到的报警器号码。

*7 ***指示检测的触点和线圈号码。

错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
7000	(1) 在多 CPU 系统里, CPU 模块错误出现在操作模式中的“PLC 的停止错误引起所有站停止”被选择的站点处。 (2) 功能版本为 A 的高性能型 QCPU 装载在多 CPU 系统。 在多 CPU 系统, 站点 1 在电源打开时产生停止错误, 同时其它站不能启动(该错误出现在站点 2 到 4)。	(1) 使用外围设备读取错误的个别信息, 检查引起 CPU 模块故障的 PLC 错误, 并去除错误。 (2) 从主基板中去掉功能版本为 A 的高性能型 QCPU。	功能版本为 B 或以后的 QCPU
7002	(1) 当多 CPU 系统在初始通信阶段时, 目标 CPU 模块没有响应。 (2) 功能版本为 A 的高性能模块 QCPU 装载在多 CPU 系统组态里。	(1) 复位 CPU 模块并再运行, 如果出现相同的问题, 智能功能模块, CPU 模块或基板有故障, 因此, 更换有故障的模块或单元。或者联系最近的三菱代理商。 (2) 从主基板中去掉功能版本为 A 的高性能型 QCPU 或者, 用功能版本 B 的高性能型 QCPU 替换功能版本为 A 的高性能型 QCPU。	
7003	当多 CPU 系统在初始通信阶段时, 目标 CPU 模块没有响应。	复位 CPU 模块并再运行, 如果出现相同的问题, 智能功能模块, CPU 模块或基板有故障, 因此, 更换有故障的模块或单元。或者联系最近的三菱代理商。	
7010	(1) 在多 CPU 系统, 故障 CPU 被装载。 (2) 功能版本为 A 的高性能型 QCPU 装载在多 CPU 系统组态里。(在功能版本为 B 的高性能模块 QCPU 上检测到一个错误。) (3) 在多 CPU 系统里, 2 到 4 号 CPU 的任一个在电源打开时被复位(该错误只出现在被复位的 CPU 模块里)。	(1) 使用外围设备读取错误的个别信息, 同时更换有故障的站。 (2) 将站的功能版本由 A 更换为 B。 (3) 不要复位从 2 到 4 号 CPU 的 CPU 模块。复位 1 号 CPU 的高性能型 QCPU 同时重启多 CPU 系统。	功能版本为 B 或以后的 QCPU
7020	在多 CPU 系统里, CPU 模块错误出现在操作模式中的“PLC 的停止错误引起所有站停止”未被选择的 CPU 模块处。 (在 CPU 故障出现的站以外的其他站点的 CPU 模块上检测到错误。)	使用外围设备读取错误的个别信息, 检查在 CPU 模块故障里 CPU 模块引起的错误, 同时去除错误。	功能版本为 B 或以后的 QCPU
9000	警报器设定为开。	使用外围设备读取错误的个别信息, 检查对应于数字值的程序(警报器号码)。	○
9010	CHK 指令检测到错误。	使用外围设备读取错误的个别信息, 检查对应于数字值(错误号码)的程序。	
9020	在自动写入到标准 ROM 过程中, 完成存储数据到 ROM。 (BOOT LED 也在闪动)。	设置参数使能驱动到标准 ROM, 再次接通电源, 并从标准 ROM 中进行引导操作。	功能版本为 B 或以后的 QCPU
10000	在多 CPU 系统, 错误出现在过程 CPU/高性能型 QCPU 以外的 CPU 模块里。	使用相应的 CPU 模块的软件包来检查出现的错误的细节。	功能版本为 B 或以后的 QCPU

* 1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

11.2.3 过程 CPU 的错误代码列表

下列信息涉及错误代码，错误信息的意义，原因和修改的措施。

在相应 CPU 栏目的“0”表示错误适用于各种类型的 CPU。

“REM”表示同远程 I/O 模块的兼容性。

错误代码 (SD0) * 1	错误信息	公共信息 (SD5 到 15) * 1	个别信息 (SD16 到 26) * 1	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
1000	MAIN CPU DOWN	—	—	关闭	闪烁/打开	停止	一直	
1001								
1002								
1003								
1004								
1005								
1006				关闭	闪烁	停止	一直	
1007								
1008								
1009								
1010	END NOT EXECUTE	—	—	关闭	闪烁	停止	当执行 END 指令时。	
1011								
1012								
1020								
1101	RAM ERROR	—	—	关闭	闪烁	停止	在电源上电/复位时	
1102								
1103								
1104								
1105								
1200	OPE. CIRCUIT ERR.	—	—	关闭	闪烁	停止	在电源上电/复位时	
1201								
1202								
1300	FUSE BREAK OFF	单元号码	—	关闭/打开	闪烁/打开	停止/继续 * 2	当执行 END 指令时。	
1310	I/O INT ERROR	—	—	关闭	闪烁	停止	在中断过程中	

* 1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

* 2 错误出现时的 CPU 操作状态能在参数下设定。(LED 显现会相应地改变)

	错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
	1000	运行模式挂起或主 CPU 的故障。 (1) 由于噪音或其它原因引起的功能故障。 (2) 硬件故障。	<ul style="list-style-type: none"> ● 采取噪音减少措施。 ● 复位 CPU 模块并再次运行。如果再次出现相同的错误, 这表示是一个 CPU 模块硬件故障。因此改变有故障的模块, 也可以与最近的三菱代理商取得联系。 	○
	1001			○
	1002			
	1003			
	1004			
	1005			○ + Rem
	1006			○
	1007			
	1008			
	1009	电源模块, CPU 模块, 主基板, 扩展基板或者扩展电缆有故障。	复位 CPU 模块并重新运行。如果再次出现相同的错误, 电源模块, CPU 模块主基板, 扩展基板或扩展电缆有故障。代理商。也可以与最近的三菱代理商取得联系。	
	1010	在没有执行 END 指令的情况下, 执行整个程序。	<ul style="list-style-type: none"> ● 采取噪音减少措施。 ● 复位 CPU 模块并再次运行。如果再次出现相同的错误, 这表示是一个 CPU 模块硬件故障。因此改变有故障的模块。也可以与最近的三菱代理商取得联系。 	○
	1011	(1) 当 END 指令执行时, 例如由于噪音被读取成另外一个代码		
	1012	(2) END 指令不知何故已经被转变成另一个指令		
	1020	一个 SFC 程序不能正常结束。 (1) SFC 程序由于噪音或其它原因不能正常结束。 (2) SFC 程序由于一些原因不能正常地结束。		
	1101	存储 CPU 顺序程序的程序内存出错。	这表示一个 CPU 模块硬件故障。因此, 需要更换有故障的模块。也可以与最近的三菱代理商取得联系。	○
	1102	RAM 里的错误被用作 CPU 工作区域。		
	1103	内部 CPU 元件错误。		
	1104	CPU 里的 RAM 地址错误。		
	1105	在 CPU 模块里的 CPU 共享内存有故障。	<ul style="list-style-type: none"> ● 采取噪音减少措施。 ● 复位 CPU 模块并再次运行。如果再次出现相同的错误, 这表示是一个 CPU 模块硬件故障。因此改变有故障的模块。也可以与最近的三菱代理商取得联系。 	○
	1200	执行 CPU 内部索引修改的线路不能正常工作。	这表示一个 CPU 模块硬件故障。因此, 需要更换有故障的模块。也可以与最近的三菱代理商取得联系。	○
	1201	内部 CPU 硬件(逻辑)不能正常工作。		
	1202	在 CPU 里执行顺序处理的线路不能正常工作。		
	1300	有一个保险丝烧坏的输出模块。	<ul style="list-style-type: none"> ● 检查输出模块的 FUSE LED, 同时当 LED 点亮时, 替换模块。 ● 有保险丝烧坏的模块也可以用外围设备来检查。监视从 SD130 到 SD137 的特殊寄存器, 并检查是否有一个位为 1, 它对应于有保险丝烧坏模块。 当显示设备连接到主基板或扩展基板时, 检查扩展电缆的链接状态和显示设备的接地状态。	○ + Rem
	1310	尽管没有中断模块, 中断也出现了。	安装的模块中的一个有硬件故障。因此, 检查安装的模块, 同时更换有故障的模块。也可以与最近的三菱代理商取得联系。	○

* 1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错 误 信 息	公 共 信 息 (SD5 到 15)	个 别 信 息 (SD16 到 26)	LED 状 态		CPU 的操 作 状 态	诊 断 时 间	
				运 行	错 误			
1401	SP. UNIT DOWN	单元号码	—	关闭/打开	闪烁/打开	停止/继续 *3	在电源上电/复位时 当智能功能模块访问时。	
1402	SP. UNIT DOWN	单元号码	程序错误位置	关闭/打开	闪烁/打开	停止/继续 *3	当智能功能模块访问指令执时。	
1403			—				当END 指令执行时。	
1411	CONTROL-BUS. ERR.	单元号码	程序错误位置	关闭	闪烁	停止	在电源上电/复位时	
1412							在FROM/TO 指令设置执行时。	
1413	CONTROL-BUS. ERR.	—	—	关闭	闪烁	停止	一直	
1414	CONTROL-BUS. ERR.	单元号码	—	关闭	闪烁	停止	当END 指令执行时。	
		—						
1415	CONTROL-BUS. ERR.	基本号	—	关闭	闪烁	停止	当END 指令执行时。	
1416		单元号码					在电源上电/复位时	
1500	AC/DC DOWN	—	—	打开	关闭	继续	一直	
1600	BATTERY ERROR	驱动号码	—	打开	打开	继续	一直	
1601				BAT. ALM LED 亮				
2000	UNIT VERIFY ERR.	单元号码	—	关闭/打开	闪烁/打开	停止/继续 *2	当END 指令执行时 当在电源上电/复位时时	

*1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

*2 错误出现时的 CPU 操作状态能在参数下设定。(LED 显现会相应地改变)

*3 通过设定参数, 每个模块的停止/继续操作是可以选择的。

错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
1401	(1) 在初始通信阶段, 智能功能模块没有响应。 (2) 智能功能模块的缓冲内存大小有错误。	这表示 CPU 模块的硬件故障。因此, 更换有故障的模块。也可以与最近的三菱代理商取得联系。	○ + Rem
1402	在程序里访问了智能功能模块, 但是没有响应。		○ + Rem
1403	(1) 当执行 END 指令时, 智能功能模块没有响应。 (2) 在智能功能模块里检测到一个错误。	访问的智能功能模块有硬件故障。因此, 更换有故障的模块。也可以与最近的三菱代理商取得联系。	○ + Rem
1411	当执行参数 I/O 分配时智能功能模块在初始通信时不能访问。在错误出现时, 相应特殊功能模块的起始 I/O 号码存储在公共信息里。	复位 CPU 模块并再次运行。如果再次出现错误, 智能功能模块, CPU 模块或基板有故障。更换有故障的模块或单元。也可以与最近的三菱取得联系。	○ + Rem
1412	FROM/TO 指令设置由于一个智能功能模块的系统总线错误不能执行。在错误出现时, 程序错误位置被存储。		○
1413	在一个多 CPU 系统组态里, 功能版本为 A 的高性能型 QCPU 被安装。	<ul style="list-style-type: none"> 从主基板中去除功能版本为 A 的高性能型 QCPU。或者, 用功能版本为 B 的高性能型 QCPU 来替换功能版本为 A 的高性能型 QCPU。或者用功能版本为 B 的过程 CPU 或高性能型 QCPU 来替换功能版本为 A 的高性能型 QCPU。 智能功能模块, CPU 模块或基板有故障。因此, 替换有故障的模块或单元。或者联系最近的三菱代理商。 	○
	在系统总线上, 检测到错误。 (1) 系统总线的自我诊断错误。 (2) CPU 模块的自我诊断错误。	复位 CPU 模块并再次运行。如果再次出现错误, 智能功能模块, CPU 模块或基板有故障。更换有故障的模块或单元。也可以与最近的三菱代理商取得联系。	○ + Rem
1414	(1) 安装模块的故障被检测到。 (2) 功能版本为 A 的高性能型 QCPU 安装在多 CPU 系统组态里。	<ul style="list-style-type: none"> 从主基板中去除功能版本为 A 的高性能型 QCPU。或者, 用功能版本为 B 的高性能型 QCPU 来替换功能版本为 A 的高性能型 QCPU。或者用功能版本为 B 的过程 CPU 或高性能型 QCPU 来替换功能版本为 A 的高性能型 QCPU。 复位 CPU 模块并再次运行。如果再次出现错误, 智能功能模块, CPU 模块或基板有故障。更换有故障的模块或单元。也可以与最近的三菱取得联系。 	○
	在系统总线上, 错误被检测到。	智能功能模块, CPU 模块, 或者基板有故障。因此, 更换有故障的模块或单元。或者, 联系最近的三菱代理商。	○ + Rem
1415	主基板或扩展基板的故障被检测到。		○
1416	在电源上电或复位时, 系统总线有故障被检测到。		○
1500	(1) 电源瞬间中断出现。 (2) 电源关闭。	检查电源。	○ Rem
1600	(1) 在 CPU 模块电池的电压降低到低于规定水平。 (2) CPU 模块电池没有连接。	<ul style="list-style-type: none"> 更换电池。 如果电池用于程序内存, 标准 RAM 或用于备份电源功能, 安装一个前连接器。 	○
1601	内存卡 1 上的电池电压降低到低于规定水平。	更换电池。	
2000	功能版本为 A 的高性能型 QCPU 安装在多 CPU 系统组态里。	用功能版本为 B 的过程 CPU 或高性能型 QCPU 来替换功能版本为 A 的高性能型 QCPU。	○
	I/O 模块信息电源上电被改变。 (1) I/O 模块(或智能功能模块) 没有, 并改正错误安装或被安装在基板上。	<ul style="list-style-type: none"> 使用外围设备读取错误公共信息, 同时检查或改变对应于数字值(模块号码)的模块。 也可以使用 GX DEVELOPER, 监视从 SD150 到 SD157 的特殊寄存器, 同时检查和替换数据位为 1 的模块的保险丝。 当显示设备连接到主基板或扩展基本板时, 检查扩展电缆的状态和显示设备的接地状态。 	○ Rem

*1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错 误 信 息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
2100	SP. UNIT LAY ERR.	单元号码	—	关闭	闪烁	停止	在在电源上电/复位时时	
2103								
2106								
2107								

*1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
2100	安装 Q160 的插槽在参数 I/O 分配中被设置为不是智能(智能功能模块)或中断(中断模块)。	用实际安装状态, 重新设定来匹配参数输入/输出指定。	○
	(1) 在参数 I/O 分配设定中, 智能功能模块被分配到一个保留给 I/O 模块的位置或反之。 (2) 在参数 I/O 分配设定中, 除了 CPU(或没有)以外的一个模块被分配到一个保留给 I/O 模块的位置或反之。 (3) 通用开关被设定给不带通用开关的模块。	<ul style="list-style-type: none"> ● 复位参数 I/O 分配设定, 使其符合智能功能模块的状态和 CPU 模块的状态。 ● 删除通用开关设置。 	
2103	(1) 2 个或更多 Q160 模块安装在单 CPU 系统里。 (2) 2 个或更多 Q160 模块在一个多 CPU 系统里, 设定给同样的控制 CPU。	<ul style="list-style-type: none"> ● 减少安装在单一 CPU 系统的 Q160 模块的数量为 1。 ● 在多 CPU 系统里将设定给相同控制的 CPUQ160 模块的数量为 1。 	○
	安装了 Q160。	卸掉 Q160。	
2106	(1) 5 个或更多 MELSECNET/H 模块安装在整个多 CPU 系统里。 (2) 5 个或更多 Q 系列以太网接口模块安装在整个多 CPU 系统里。	<ul style="list-style-type: none"> ● 在整个多 CPU 系统, 减少 MELSECNET/H 的数量为 4 或更少。 ● 在整个多 CPU 系统, 减少 Q 系列以太网模块数为 4 或更少。 	○ + Rem
	(1) 5 个或更多 MELSECNET/H 模块被安装。 (2) 5 个或更多 Q 系列以太网接口模块被安装。 (3) 在 MELSECNET 网络系统里, 存在相同的网络号码或站号码。	<ul style="list-style-type: none"> ● 减少 MELSECNET/H 模块的数量为 4 或更少。 ● 减少 Q 系列以太网模块的数量为 4 或更少。 ● 检查网络号码和站号码。 	
2107	在参数 I/O 分配设定中设定的起始 X/Y 也是另一个模块的起始 X/Y。	复位参数 I/O 设定, 使其符合特殊功能模块的实际状态。	○ Rem

* 1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
2110	SP. UNIT ERROR	单元号码	程序错误位置	关闭/打开	闪烁/打开	停止/继续 *2	指令执行时。	
2111								
2112	SP. UNIT ERROR	单元号码	程序错误位置	关闭/打开	闪烁/打开	停止/继续 *2	指令执行时。 STOP → RUN	
2113		FFFF _H (固定)						
2114	SP. UNIT ERROR	单元号码	程序错误位置	关闭/打开	关闭/打开	停止/继续	指令执行时。	
2115								
2116								
2117								
2118								
2120	SP. UNIT LAY ERR.	—	—	关闭	闪烁	停止	在电源上电/复位时	
2121								
2122								
2124								
2125								
2126	SP. UNIT LAY ERR.	单元号码	—	关闭	闪烁	停止	在电源上电/复位时	
2150	SP. UNIT VER. ERR.	单元号码	—	关闭	闪烁	停止	在电源上电/复位时	
2200	MISSING PARA.	文件名称/驱动名称	—	关闭	闪烁	停止	在电源上电/复位时	
2210	BOOT ERROR	文件名称/驱动名称	—	关闭	闪烁	停止	在电源上电/复位时	
2300	ICM. OPE. ERROR	文件名称/驱动名称	—	关闭/打开	闪烁/打开	停止/继续 *2	当内存卡插入或取出时。	
2301								
2302								
2400	FILE SET ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	在电源上电/复位时	

*1 在圆括号()里的字符表示存储个别信息的特殊寄存器的号码。

*2 错误出现时的 CPU 操作状态能在参数下设定。(LED 显现会相应地改变)

	错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
	2110	使用目标为 CPU 共享内存的指令来指定的站未安装。	使用 GX DEVELOPER 读取错误的个别信息, 检查对应于值(程序错误位置)的程序, 同时进行修改。	○
		(1) 由 FROM/TO 指令设置指定的位置不是智能功能模块。 (2) 被访问的智能功能模块, 网络模块有故障。	使用 GX DEVELOPER 读取错误个别信息, 然后检查对应于数字值的 FROM/TO 指令设置(程序错误位置), 在必要时进行修改。 被访问的智能功能模块有硬件故障。因此, 更换有故障的硬件。或者, 联系最近的 三菱代理商。	
	2111	由一个链接直接元件(J□\□)指定的位置不是网络模块。		
	2112	(1) 在智能功能模块专用指令中指定的模块不是一个智能功能模块或者, 它不是相应的智能功能模块。 (2) 在网络专用指令中指定的网络号码不存在, 或者继电器目标网络不存在。 (3) 或者, 它不是相关的智能功能模块。	使用 GX DEVELOPER 读取错误的个别信息, 然后检查和编辑对应于数字值(程序错误位置)的特殊功能模块(网络模块)专用指令。	○ + Rem
	2113	在网络专用指令中指定的模块不是网络模块。		
	2114	一个在执行时用于指定其它站的指令被用做指定主站。(一个不允许主站被指定的指令)	使用 GX DEVELOPER 读取错误的个别信息, 检查对应于值(程序错误位置)的程序, 同时进行修改。	○
	2115	一个在执行时用于指定主站的指令被用做指定其它站。(一个不允许其它站被指定的指令)		
	2116	一个不允许受控于其它站的模块被指定的指令被用于类似的任务。	使用 GX DEVELOPER 读取错误的个别信息, 检查对应于值(程序错误位置)的程序, 同时进行修改。	
	2117	一个多 CPU 系统专用指令无法指定的 CPU 模块被指定。		
	2118	在多 CPU 系统里, 当在线模块变化设置被设定为使能过程 CPU 的参数, 受控于另一 CPU 的智能功能模块在 FROM 指令/智能功能模块元件(L□\G□)里已经被指定。	<ul style="list-style-type: none">● 在多 CPU 系统, 当过程 CPU 的在线模块改变被执行时, 修改程序使得不能访问到受控于另一 CPU 的模块。● 在多 CPU 系统, 当访问到受控于另一 CPU 的模块时, 设置在线模块变化设置, 使得过程 CPU 的参数无效。	
2120	QA1S□B 或 QA□B 其中的一个被连接。	使用一个 Q□B。	○ + Rem	
2121	CPU 模块被安装在 CPU 插槽以外的插槽或从 0 到 2 的插槽上。	检查 CPU 模块的安装位置, 同时在, 并改正错误的插槽重新安装 CPU。		
2122	QA1S□B 用作主基板。	使用 Q3□B 作为主基板。		
2124	(1) 模块被安装在 65 号或更高的插槽。 (2) 模块安装在用基板分配设定指定的插槽数之后的插槽上。 (3) 模块安装在 4096 号点之后的 I/O 点。 (4) 安装在 4096 点的模块占据较高点。	<ul style="list-style-type: none">● 去除安装在 65 号或更高插槽的模块。● 去除安装在用基板分配设定指定的插槽数之后的插槽的模块。● 去除安装在 4096 号点之后的 I/O 点的模块。● 改变最后的模块到不超过 4096 点的模块。		
2125	(1) 高性能型 QCPU 不能辨认的模块被安装。 (2) 智能功能模块没有响应。	<ul style="list-style-type: none">● 安装可以使用的模块。● 智能功能模块有硬件故障。因此, 更换有故障的硬件。或者, 联系最近的 三菱代理商。		
2126	在多 CPU 系统里的 CPU 模块位置不是下列中的一个。 (1) 在 CPU 模块的左侧有一个空插槽。 (2) 除了高性能型 QCPU/过程 CPU 外的模块, 例如运动控制器, 或 PC CPU 模块被安装在高性能型 QCPU/过程 CPU 的左侧。	<ul style="list-style-type: none">● 去掉在 CPU 模块之间的空插槽。(将空插槽设在 CPU 模块的右侧)● 去除不是高性能型 QCPU 模块的模块, 安装在高性能型 QCPU 模块之间的 QCPU 模块, 用高性能型 QCPU 来填充这些插槽。 在高性能型 QCPU 的右侧安装运动控制器。	○	
2150	在多 CPU 系统, 同多 CPU 系统不兼容的智能功能模块的控制 CPU 被设定为不是 1 号 CPU。	<ul style="list-style-type: none">● 改变智能功能模块为与多 CPU 系统兼容的。(功能版本为 B)● 改变同多 CPU 系统不兼容的智能功能模块的控制 CPU 的设置为 1 号 CPU。		
2200	通过 DIP 开关指定的驱动上没有参数文件被作为有效驱动。	<ul style="list-style-type: none">● 检查和修改参数激活驱动开关的设置。● 在通过参数激活驱动开关指定的驱动上放置参数文件。		
	2210	源文件的内容不正确。	检查引导设置。	
	2300	(1) 在没有将内存卡 IN/OUT 开关切换为 OFF 下, 去除内存卡。 (2) 尽管内存卡没有实际安装, 内存卡的 IN/OUT 开关打开为 ON。	<ul style="list-style-type: none">● 在将内存卡 IN/OU 开关切换为 OFF 后, 去除内存卡。● 在插入内存卡后, 打开卡插入开关。	○
		(3) 内存卡没有被格式化。 (4) 内存卡格式状态不正确。		
	2302	安装了不同 CPU 模块一起被使用的内存。	检查内存卡。	
	2400	在参数里的 PLC 文件设定处指定的文件不能找到。	使用 GX DEVELOPER 读取错误的个别信息, 检查对应于数字值(参数号码)的参数驱动名称和文件名称, 并改正错误。 在参数里建立文件设定, 并写入到 CPU 模块。	○

* 1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
2401	FILE SET ERROR	文件名称/驱动名称	参数编号	关闭	闪烁	停止	在电源上电/复位时	
2410	FILE OPE. ERROR	文件名称/驱动名称	程序出错地址	关闭/打开	闪烁/打开	停止/继续 *2	指令执行时。	
2411								
2412								
2413								
2500	CAN'T EXE. PRG.	文件名称/驱动名称	—	关闭	闪烁	停止	在电源上电/复位时	
2501								
2502								
2503								
2504								
3000	PARAMETER ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	在电源上电/复位时 STOP→RUN	
3001								
3002								
3003	PARAMETER ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	执行 END 指令。	
	PARAMETER ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	在电源上电/复位时 STOP→RUN	
3004	PARAMETER ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	在电源上电/复位时 STOP→RUN	
3010	PARAMETER ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	在电源上电/复位时 STOP→RUN	
3012								
3013								
3014								

*1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

*2 错误出现时的 CPU 操作状态能在参数下设定。(LED 显现会相应地改变)

错误代码 (SD0) *1	错误内容和原因	修改方法	相应 CPU
2401	通过执行引导操作和自动写入标准 ROM，超出程序内存容量。	<ul style="list-style-type: none"> ● 检查和修改参数(引导设置)。 ● 删除不必要的程序内存中的文件。 ● 在参数里对启动选择“清除程序内存”，使得在程序内存清除后启动开始。 	○
	在参数 PLC RAS 设定的故障历史区域指定的文件没有被创建。	使用 GX DEVELOPER 读取错误的个别信息，检查对应于数字值(参数号码)的参数驱动名称和文件名称，并改正错误。 检查在内存卡的剩余空间。	
2410	在程序内存里不存在指定文件， 改错误在 ECALL, EFCALL, PSTOP, PSCAN, POFF 或 PLOW 指令执行时可能出现。	使用 GX DEVELOPER 读取错误的个别信息，检查对应于数字值的参数驱动名称和文件名称(参数号码)，并改正错误。 在参数设定里建立文件，写入到 CPU 模块。	○
2411	指定文件存在于程序内存，但是在参数对话框的程序设置里没有注册。 该错误在 ECALL, EFCALL, PSTOP, PSCAN, POFF 或 PLOW 指令执行时可能出现。	使用 GX DEVELOPER 读取错误的个别信息，检查对应于数字值(参数号码)的参数驱动名称和文件名称，并改正错误。	
2412	SFC 程序文件是顺序程序不能指定的文件。		
2413	没有数据写入到顺序程序指定的文件。	使用 GX DEVELOPER 读取错误的个别信息，检查对应于数字值(参数号码)的参数驱动名称和文件名称，并改正错误。 检查以确定指定文件没有写保护。	
2500	(1) 这是一个程序文件，它使用的元件超出了在 PLC 参数元件设定里设置的范围。 (2) 在 PLC 参数设定改变后，只有参数写进 PLC。	(1) 用 GX DEVELOPER 读取公共信息，同时检查从而确信对应于数字值(文件名称)的参数元件分配设置和程序文件元件分配，如有必有可以修改。 (2) 如果 PLC 参数元件设定改变，成批写入参数和程序文件进入 PLC。	○
2501	尽管没有在参数程序设置中被设定，仍然有多程序文件。	编辑参数程序设置为“是”。 或者，删除不必要的文件。	○
2502	程序文件不正确。 或者，文件内容不是顺序程序的内容。	检查是否程序版本是否是 * * *.QPG，同时检查内容确认为顺序程序。	
2503	没有程序文件	● 检查程序构成。	
2504	2 个或更多 SFC 常规程序或控制程序被指定。	● 检查参数和程序构成。	
3000	在多 CPU 系统，受控于另一站的智能功能模块在参数的中断指针设置中被指定。 用于计时器的时间限定设置，RUN-PAUSE 触点，公共指针号，一般数据处理，空插槽数，或系统中断设定的参数设定超出了能被 CPU 模块使用的范围。	<ul style="list-style-type: none"> ● 指定受控于主站的智能功能模块的起始 I/O 号码。 ● 删除参数的中断指针设定。 	○
3001	参数内容已经被破坏。	● 使用 GX Developer 时，读取错误的细节信息，检查对应于那些数字值(参数号码)的参数条目，同时在必要时修改。	○ + Rem
3002	在 PLC 文件设定参数里，当选择“使用下列文件”作为文件寄存器时，尽管文件寄存器容量已经设置，但指定文件不存在。	● 在修改参数设定后，错误仍然存在，可能的原因是 CPU 模块的程序内存或内存卡的内存错误。因此，改变有故障的模块。或者，联系最近的三菱代理商。	
3003	多 CPU 系统的自动刷新范围超过了文件寄存器容量。	在所有范围中，更改文件寄存器文件为刷新使能。	○
	在参数元件设置设定的元件数超过了可能的 CPU 模块范围。	<ul style="list-style-type: none"> ● 使用 GX Developer 时，读取错误的细节信息，检查对应于那些数字值(参数号码)的参数条目，同时在必要时修改。 ● 在修改参数设定后，错误仍然存在，可能的原因是 CPU 模块的程序内存或内存卡的内存错误。因此，改变有故障的模块。或者，联系最近的三菱代理商。 	
3004	参数文件不正确。 或者，文件内容不是参数。	检查是否参数文件版本是 * * *.QPA，同时检查文件内容确信是参数。	
3010	CPU 模块的参数设定号码不同于在多 CPU 系统的实际号码。	用实际安装的 CPU 的号码来匹配(多 CPU 设置的现有量)。(在 I/O 分配中的 CPU(空闲)设置)。	○
3012	多 CPU 设定或控制 CPU 设置不同于在多 CPU 系统中的参考站的设置。	用参考站的设置(1 号站)来匹配在参数里的多 CPU 设置或控制 CPU 设置。	
3013	在多 CPU 系统里，多 CPU 自动刷新设置是下列之一。 (1) 当位元件指定为刷新元件时，不是 16 倍数的号码被指定为刷新启动元件。 (2) 指定元件不能被指定的元件。 (3) 传送点号码是奇数。	在多 CPU 自动刷新参数里检查下列情况，同时修改。 <ul style="list-style-type: none"> ● 当指定位元件时，将刷新启动元件指定为 16 的倍数。 ● 指定能被指定为刷新元件的元件。 ● 设定传送点号码为偶数。 	
3014	<ul style="list-style-type: none"> ● 被用作标准的单元，同在多 CPU 系统结构中的在线模块变化参数(多 CPU 系统参数)里设置的内容不相同。 ● 在多 CPU 系统结构中，尽管安装的 CPU 不支持在线模块变化参数，在线模块变化设置却被使能。 	<ul style="list-style-type: none"> ● 确信在线模块变化参数同用作标准的单元的机器号码相匹配。 ● 如果不支持在线模块变化参数的 CPU 模块被安装，用支持在线模块变化参数的 CPU 来替换。 	○

*1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
3100	Link PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	当在电源上电/复位时 S10P→RUN	
3101	Link PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	当执行 END 指令时。	
	Link PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	当在电源上电/复位时 S10P→RUN	
3102	Link PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	当在电源上电/复位时 S10P→RUN 当执行 END 指令时。	
3103	Link PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	当在电源上电/复位时 S10P→RUN	
3104								
3105								
3106	Link PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	当执行 END 指令时。	
	Link PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	当在电源上电/复位时 S10P→RUN	
3107	Link PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	当在电源上电/复位时 S10P→RUN	
3200	SFC PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	STOP→RUN	
3201								
3202								
3203								
3300	SP. PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	当在电源上电/复位时 S10P→RUN	

*1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
3100	在多 CPU 系统, 受控于另一站的 MELSECNET/H 在 MELSECNET/H 的网络设置参数里被指定为起始 I/O 号码。 在普通站运行的 MELSECNET/H 的网络参数被写到控制站, 或者在控制站运行的 MELSECNET/H 的网络参数被写到普通站点。(网络参数通过复位反映在模块端)	(1) 删除受控于另一站的 MELSECNET/H 的 MELSECNET/H 网络参数。 (2) 改变受控于主机站的 MELSECNET/H 的起始 I/O 设置。 复位 CPU 模块。	○
	(1) 实际安装模块的号码不同于在 MELSECNET/H 的模块设置参数中指定的号码。 (2) 实际安装模块的起始 I/O 号码不同于在 MELSECNET 的网络参数指定的号码。 (3) 参数里的一些数据不能处理。 (4) 当电源上电时, MELSECNET 的站类型被改变。(改变站类型需要 RESET → RUN)	(1) 检查网络参数和安装状态, 如果有所区别, 匹配网络参数和安装状态。如果任何网络参数被修改, 写入 CPU 模块。 (2) 确认扩展基板的扩展级的数量的设定。 (3) 检查扩展基板和连接器的连接状态, 当显示设备连接到主基板或扩展基板时, 检查连接状态。 (4) 如果在以上 1 到 3 的检查执行后, 依然有错误出现, 表示是硬件故障。因此, 更换有故障的硬件。或者, 联系最近的三菱代理商。	
	链接刷新范围超过了文件寄存器的容量。	改变文件寄存器为在整个范围内可以刷新的文件。	
	(1) 当 MELSECNET/H 模块的站数为 0 时, 内部 PLC 网络参数设置已经建立。 (2) 当 MELSECNET/H 模块的站数不是 0, 远程主站参数设置已经设定。	在参数里修改 MELSECNET/H 模块的类型或站号码来符合使用的系统。	
3101	通过参数指定的网络号码不同于实际安装网络的号码。 通过参数指定的起始 I/O 号码不同于实际安装 I/O 单元的号码。 通过参数指定的网络等级不同于实际安装的网络等级。 MELSECNET/H 的网络刷新参数超出指定区域。	(1) 检查网络参数和安装状态, 如果有所区别, 匹配网络参数和安装状态。如果任何网络参数被修改, 写入 CPU 模块。 (2) 确认扩展基板的扩展级的号码设定。 (3) 检查扩展基板和连接器的连接状态, 当显示设备连接到主基板或扩展基板时, 检查连接状态。 (4) 如果入上检查 1 和 3 执行后, 依然有错误出现, 表示是硬件故障。因此, 更换有故障的硬件。或者, 联系最近的三菱代理商。	○
	一个多远程 I/O 网络由使用了不支持 MELSECNET/H 的多远程 I/O 网络的模块而构成。	使用支持 MELSECNET/H 多远程 I/O 网络的模块。	
3102	在网络模块执行网络参数检查时, 发现一个错误。	<ul style="list-style-type: none"> 在修改网络参数后写入。 如果修改后错误依然出现, 这表示是硬件故障。因此, 更换有故障的模块。或者, 联系最近的三菱代理商。 	
3103	在多 CPU 系统, 受控于另一站的 Q 系列以太网接口模块被指定为以太网设置参数的起始 I/O 号码。	<ul style="list-style-type: none"> 删除受控于另一站的 Q 系列以太网接口模块的以太网设置参数。 将设置该为受控于主机站的 Q 系列以太网接口模块的第一个 I/O 号码。 	○
	(1) 尽管在参数里以太网模块的数量设定为 1 或更多, 单元的实际安装数量为 0。 (2) 以太网模块设定参数的起始 I/O 号码不同于实际安装模块的号码。	<ul style="list-style-type: none"> 在修改网络参数后写入。 如果修改后错误依然出现, 这表示是硬件故障。因此, 更换有故障的模块。或者, 联系最近的三菱代理商。 	○ Rem
3104	(1) 以太网和 MELSECNET/10 使用同样的网络号码。 (2) 通过参数设定的网络号码, 站号码或组号码超出范围。 (3) I/O 号码超出范围。	<ul style="list-style-type: none"> 在修改网络参数后写入。 如果修改后错误依然出现, 这表示是硬件故障。因此, 更换有故障的模块。或者, 联系最近的三菱代理商。 	○ + Rem
3105	在多 CPU 系统, 受控于另一站的 Q 系列 CC-link 链接模块被指定为 CC-link 链接设定参数的起始 I/O 号码。	<ul style="list-style-type: none"> 删除受控于另一站的 Q 系列以太网接口模块的以太网设置数。 改变设置成为受控于主站的 Q 系列以太网接口模块的最初 I/O 号码。 	○
	(1) 尽管在网络参数里设定的 CC-Link 链接模块的数量为 1 或更多, 实际安装的单元数量为 0。 (2) 在普通参数的起始 I/O 号码不同于实际安装模块的号码。 (3) 对于 CC-Link 链接模块数量设定参数的站点等级不同于实际安装站点的等级。	<ul style="list-style-type: none"> 在修改网络参数后写入。 如果在修改后仍发生错误, 则表示是硬件故障, 因此, 应更换有故障的模块, 或者, 联系最近的三菱代理商。 	○ + Rem
3106	CC-Link 链接刷新范围超出文件寄存器容量。	改变文件寄存器为在整个范围内允许刷新的文件。	○
	对于 CC-Link 链接的网络刷新参数超出范围。	检查参数设置。	○ + Rem
3107	CC-Link 链接参数的内容不正确。	检查参数设置。	○ + Rem
3200	参数设定非法 (1) 尽管在 PLC 参数对话框的 SFC 设置里块 0 设置为“自动启动”, 但块 0 不存在。	使用 GX Developer 读取错误的公共信息, 检查对应于数字值(程序错误位置)的错误步, 同时改正问题。	○
3201	块参数设定非法。		
3202	在参数中指定的步继电器的数量少于通过程序使用的数量。		
3203	在参数里对于 SFC 程序设定的执行类型是扫描执行类型。		
3300	在 GX Configurator 设定的智能功能模块参数的起始 I/O 号码不同于实际的 I/O 号码。	检查参数设置。	○ + Rem

* 1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
3301	SP. PARA. ERROR	文件名称	参数号码	关闭	闪烁	停止	当执行 END 指令时。	
3302							在电源上电/复位时 STOP → RUN	
3303	SP. PARA. ERROR	文件名称/驱动名称	参数号码	关闭	闪烁	停止	在电源上电/复位时 STOP → RUN	
3400	REMOTE PASS. ERROR	—	—	关闭	闪烁	停止	在电源上电/复位时 STOP → RUN	
3401								
4000	INSTRCT CODE ERR.	程序错误位置	—	关闭	闪烁	停止	在电源上电/复位时 STOP → RUN	
4001								
4002								
4003								
4004	INSTRCT CODE ERR.	程序错误位置	—	关闭	闪烁	停止	在电源上电/复位时 STOP → RUN	
4010	MISSING END INS.	程序错误位置	—	关闭	闪烁	停止		
4020	CAN' T SET (P)	程序错误位置	—	关闭	闪烁	停止		
4021								
4030	CAN' T SET (I)	程序错误位置	—	关闭	闪烁	停止		
4100	OPERATION ERROR	程序错误位置	—	关闭/打开	闪烁/打开	停止/继续 *2		
4101						停止/继续 *2		
4102		程序错误位置	—			停止/继续 *2		
4107		程序	程序错误位置			停止/继续 *2		
4200	FOR NEXT ERROR	程序错误位置	—	关闭	闪烁	停止	指令执行时。	
4201	FOR NEXT ERROR	程序错误位置	—	关闭	闪烁	停止	指令执行时。	
4202								
4203								

*1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

*2 错误出现时的 CPU 操作状态能在参数下设定。(LED 显现会相应地改变)

错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
3301	智能功能模块的刷新设置超出文件寄存器的容量。	改变文件寄存器为在整个范围内可以刷新的文件。	○
	智能功能模块的刷新参数设定在现有范围之外。	检查参数设置。	○ + Rem
3302	智能功能模块的刷新参数不正常。	检查参数设置。	○
3303	在多 CPU 系统, 自动刷新设置和相似参数设置被设定到在另一站的控制下建立的智能功能模块。	<ul style="list-style-type: none"> ● 删除受控于另一站的智能功能模块的自动刷新设置或相似参数设置。 ● 将设置该为受控于主机站的智能功能模块的自动刷新设置或相似设置。 	○
3400	在远程口令文件里的目标模块的起始 I/O 号码设定不在 0H 到 0FF0H 之内。	目标模块的第一个 I/O 号码设定到 0H 到 0FF0H 之间。	
3401	指定为远程口令文件的第一个 I/O 号码的位置由于下列的某个原因不准确。 (1) 模块没有安装。 (2) QJ71C24(-R2) 或 Q 系列以太网接口模块以外的智能功能模块。 (3) 功能版本为 A 的 QJ71C24(-R2) 或 Q 系列以太网接口模块。 受控于另一站的功能版本为 B 的 QJ71C24(-R2) 或 Q 系列以太网接口模块在多 CPU 系统内指定。	在远程口令文件的起始 I/O 号码指定的位置安装功能版本为 B 的串行通信模块, 以太网模块或调制解调器接口模块。 <ul style="list-style-type: none"> ● 将设置改为在本地站的控制下的功能版本为 B 的 QJ71C24(-R2) 或 Q 以太网接口模块。 ● 删除远程密码设定。 	
4000	(1) 程序包含一个不能解码的指令号码。 (2) 一个不能使用的指令包括在程序里。	使用外围设备读取错误的公共信息, 检查对应于数字值(程序错误位置)的错误步, 同时改正问题。	○
4001	尽管它不是一个 SFC 程序, 程序包含一个 SFC 专用的指令。		○ + Rem
4002	通过程序指定的扩展指令有一个错误的指令名称。		
4003	通过程序指定的扩展指令有个错误的元件数量。		
4004	程序所指定的扩展指令有个无法使用的元件。	使用外围设备读取错误的公共信息, 检查对应于数字值(程序错误位置)的错误步, 同时改正问题。	○
4010	在程序里没有 END(FEND) 指令。	使用外围设备读取错误的公共信息, 检查对应于数字值(程序错误位置)的错误步, 同时改正问题。	○
4020	被程序使用的内部文件指针的总数超过了参数里设置的内部文件指针的数量。		
4021	由单个文件使用的通用指针号码重叠。		
4030	由单个文件使用的分配指针号码重叠。		
4100	指令不能处理包含的数据。	使用外围设备读取错误的公共信息, 检查对应于数字值(程序错误位置)的错误步, 同时改正问题。	○
4101	(1) 通过指令处理的数据的指定元件号码超出了使用范围。 或者, 通过指令指定的元件存储的数据或常量超过使用范围。		
4102	在多 CPU 系统, 链接直接元件(J□\G□)被指定为在另一站控制下的网络模块。	<ul style="list-style-type: none"> ● 从程序里删除在另一站控制下指定网络模块的链接直接元件。 ● 使用链接直接元件, 在本地站的控制下指定能够网络模块。 	○
	用专用网络指令指定的网络号码和站号码不正确。 链接直接元件(j□\w□)没有正确设定。 通过扩展指令指定的模块号码/网络号码/字符串数超出了设定范围。 通过扩展指令指定的字符串不能使用。	使用外围设备读取错误的公共信息, 检查对应于数字值(程序错误位置)的错误步, 同时改正问题。	○ + Rem
4107	33 个或更多 CPU 专用指令在一个高性能型 QCPU 上执行。	使用多 CPU 专用指令完成位, 提供互锁来防止一个 CPU 执行 32 个或更多的多 CPU 专用指令。	○
4200	在 FOR 指令执行后, 没有 NEXT 执行。 或者, NEXT 指令数少于 FOR 指令。	使用外围设备读取错误的公共信息, 检查对应于数字值(程序错误位置)的错误步, 同时改正问题。	○
4201	尽管没有 FOR 指令被执行, NEXT 仍然执行。 或者, NEXT 指令数多于 FOR 指令。	使用外围设备读取错误的公共信息, 检查对应于数字值(程序错误位置)的错误步, 同时改正问题。	○
4202	多于 16 级嵌套被处理。	保持嵌套级数为 16 或更低。	
4203	尽管先前没有 FOR 指令的执行, BREAK 指令被执行。	使用外围设备读取错误的公共信息, 检查对应于数字值(程序错误位置)的错误步, 同时改正问题。	

* 1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
				运行	错误			
4210	CAN' T EXECUTE (P)	程序错误位置	—	关闭	闪烁	停止	当指令执行时。	
4211								
4212								
4213								
4220	CAN' T EXECUTE (I)	程序错误位置	—	关闭	闪烁	停止	当指令执行时。	
4221								
4223								
4230	INST. FORMAT ERR.	程序错误位置	—	关闭	闪烁	停止	当指令执行时。	
4231								
4235								
4400	SFCP. CODE ERROR	程序错误位置	—	关闭	闪烁	停止	停止→运行	
4410	CAN' T SET (BL)	程序错误位置	—	关闭	闪烁	停止	停止→运行	
4411								
4420	CAN' T SET (S)	程序错误位置	—	关闭	闪烁	停止	停止→运行	
4421								
4422								
4500	SFCP. FORMAT ERR.	程序错误位置	—	关闭	闪烁	停止	停止→运行	
4501								
4502								
4503								
4504								
4600	SFCP. OPE. ERROR	程序错误位置	—	关闭/打开	闪烁/打开	停止/继续 *2	当指令执行时。	
4601								
4602								
4610	SFCP. EXE. ERROR	程序错误位置	—	打开	打开	继续	停止→运行	
4611								
4620	BLOCK EXE. ERROR	程序错误位置	—	关闭	闪烁	停止	当指令执行时。	
4621								
4630	STEP EXE. ERROR	程序错误位置	—	关闭	闪烁	停止	当指令执行时。	
4631								
4632								
4633								
5000	WDT ERROR	时间(设定值)	时间(实际测量值)	关闭	闪烁	停止	一直	
5001								
5010	PRG. TIME OVER	时间(设定值)	时间(实际测量值)	打开	打开	继续	一直	
5011								

*1 在圆括号()里的字符表示存储个别信息的特殊寄存器的号码。

*3 只能在剩余系统里被检测到。能在控制系统或备用系统里被检测到。

*4 只能在剩余系统的控制系统里被检测到。

*5 只能在剩余系统的备用系统里被检测到。

	错误代码 (SD0) * 1	错误内容和原因	修改方法	相应 CPU
	4210	CALL 指令执行时，在指定指针处没有子函数。	使用 GX Developer 读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	○
	4211	在执行的子函数程序里，没有 RET 指令		
	4212	在主程序中，RET 指令在 FEND 指令之前。		
	4213	多于 16 级嵌套被处理。		
	4220	尽管出现中断输入，但相应中断指针不存在。	使用 GX Developer 读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	○
	4221	在运行的中断程序里 RET 指令不存在。		
	4223	在主程序里，IRET 指令在 FEND 指令前有。		
	4230	CHK 和 CHKEND 指令的数量不相等。		
	4231	IX 和 IXEND 指令的数量不相等。		
	4235	对于 CHK 指令的检查条件组态不正确。 或者，在一个低速执行类型程序里，CHK 指令被使用。		
	4400	在 SFC 程序里没有 SFCP 或 SFCPEND 指令。	使用 GX Developer 读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	○
	4410	通过 SFC 程序指定的块号码超出了范围。		
	4411	块号码指定在 SFC 程序里重叠。		
	4420	在 SFC 程序里指定的步数超过了 511。		
	4421	在所有 SFC 程序的总步数超过范围。	将总的步数减少到最大值之下。	
	4422	步数指定在 SFC 程序里重叠。	使用 GX Developer 读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	
	4500	在 SFC 程序里的 BLOCK 和 BEND 指令的数量不相等。	使用 GX Developer 读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	○
	4501	在 SFC 程序里从 STEP * 到 TRAN * 到 TSET 到 SEND 指令的组态不正确。		
	4502	在 SFC 程序块里没有 STEPI * 指令。		
	4503	在 SFC 程序里通过 TSET 指令中指定的步不存在。		
	4504	在 SFC 程序里通过 TAND 指令中指定的步不存在。		
	4600	SFC 程序里包含不能处理的数据。	使用 GX Developer 读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	○
	4601	超出了 SFC 程序能指定的元件范围。	程序自动受制于一个初始化启动。	
	4602	在 SFC 程序里的 START 指令由 END 指令来处理。		
	4610	SFC 程序的假定开始的活动步的信息不正确。	通过 GX Developer 读取出错公用信息，检查数值(程序错误地址)对应的出错步，并纠正错误。 此程序自动受制于一个初始化启动。	
	4611	当为 SFC 程序指定假定开始时，钥匙开关在 RUN 时被复位。		
	4620	启动在 SFC 程序里已经启动的块上执行。	复位 CPU 模块并再运行，如果出现相同的问题，智能功能模块，CPU 模块或基板有故障，因此，改变有故障的模块或单元。或者联系最近的三菱代理商。	○
	4621	启动在 SFC 程序里不存在的块上尝试。		
	4630	启动在 SFC 程序里已经启动的块上执行。	使用 GX Developer 读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	
	4631	启动在 SFC 程序里不存在的块上尝试。	复位 CPU 模块并再运行，如果出现相同的问题，智能功能模块，CPU 模块或基板有故障，因此，改变有故障的模块或单元。或者联系最近的三菱代理商。	
	4632	能被 SFC 程序指定的块上有许多相同活动步。	使用 GX Developer 读取错误的公共信息，检查对应于数字值(程序错误位置)的错误步，同时改正问题。	
	4633	在被指定的块上有很多相同活动步。		
	5000	对于初始执行类型文件的程序扫描时间超过了 PLC RAS 参数里设定的初始执行 WDT 时间设置。	使用 GX DEVELOPER 读取错误的个别信息，检查数字值，同时如果必要，缩短扫描时间。	○
	5001	程序扫描时间超过了设定在参数 PLC RAS 参数中的 WDT 值。		
	5010	(1) 程序扫描时间超出在 PC RAS 设定参数中指定的连续扫描设定时间。 (2) PC RAS 设定参数中指定的低速程序执行时间超出固定扫描极限时间。	(1) 检查恒定扫描设置时间。 (2) 检查在参数里的恒定扫描时间和低速程序执行时间使得恒定扫描的多余时间能被完全保留。	○
	5011	低速扫描类型程序的扫描时间超过了在参数 PC RAS 设定中设定的低速执行 WDT。	使用 GX DEVELOPER 读取错误的个别信息，检查数字值，同时如果必要，缩短扫描时间。	

* 1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

错误代码列表(待续)

错误代码 (SD0) *1	错误信息	公共信息 (SD5 到 15)	个别信息 (SD16 到 26)	LED 状态		CPU 的操作 状态	诊断时间	
7000	MULT CPU DOWN	单元/模块号码	—	关闭	闪烁	停止	一直	
7002							在电源上电/复位时	
7003							在电源上电/复位时	
7010	MULTI EXE. ERROR	单元/模块号码	—	关闭	闪烁	停止	在电源上电/复位时	
7020	MULTI CPU ERROR	单元/模块号码	—	打开	打开	继续	一直	
9000	F**** *6	程序错误位置	警报器号码	打开 USER LED On	关闭 USER LED On	继续	当执行指令时。	
9010	<CHK> ERR ***-*** *7	程序错误位置	过失号	打开 USER LED On	关闭 USER LED On	继续	当执行指令时。	
9020	BOOT OK	—	—	关闭	闪烁	停止	在电源上电/复位时	
10000	CONT. UNIT ERROR	—	—	—	—	—	—	

*1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

*6 ****表示检测警报器号码。

*7 ***指示检测的触点和线圈号码。

错误代码 (SD0) *1	错误内容和原因	修改方法	相应 CPU
7000	(1) 在多 CPU 系统里, CPU 故障出现在操作模式的“PLC 的停止错误使所有站停止”被选择的站点处。 (2) 功能版本为 A 的高性能型 QCPU 安装在多 CPU 系统里。 在多 CPU 系统, 站点 1 在电源上电时产生停止错误, 同时其它站不能启动(该错误出现在站 2 到 4)。	(1) 使用外围设备读取错误的个别信息, 检查引起 CPU 模块故障的 CPU 的错误, 同时去除错误。 (2) 从主基板中去掉功能版本为 A 的高性能型 QCPU。 使用外围设备读取错误的个别信息, 检查引起 CPU 模块故障的 CPU 的错误, 同时去除错误。	○
7002	(1) 当多 CPU 系统在初始通信阶段时, 从目标 CPU 模块没有响应。 (2) 功能版本为 A 的高性能型 QCPU 安装在多 CPU 系统组态里。	(1) 复位 CPU 模块并再运行, 如果出现相同的问题, 智能功能模块, CPU 模块或基板有故障, 因此, 改变有故障的模块或单元。或者联系最近的三菱代理商。 (2) 从主基板中去掉功能版本为 A 的高性能型 QCPU。或者, 用功能版本 B 的高性能型 QCPU 替换功能版本为 A 的高性能型 QCPU。	
7003	当多 CPU 系统在初始通信阶段时, 从目标 CPU 模块没有响应。	复位 CPU 模块并再运行, 如果出现相同的问题, 智能功能模块, CPU 模块或基板有故障, 因此, 改变有故障的模块或单元。或者联系最近的三菱代理商。	
7010	(1) 在多 CPU 系统, 有故障的 CPU 被安装 (2) 功能版本为 A 的高性能型 QCPU 安装在多 CPU 系统组态里。 (在功能版本为 B 的高性能型 QCPU 上检测到一个错误。) (3) 在多 CPU 系统里, 2 到 4 号 CPU 的任意一个在电源上电时被复位。 (该错误只出现在被复位的 CPU 模块里)	(1) 使用外围设备读取错误的个别信息, 同时更换有故障的站。 (2) 将功能版本为 A 的站改为功能版本为 B 的站。 (3) 不要复位从 2 号到 4 号 CPU 的 CPU 模块。复位 1 号 CPU 的高性能型 QCPU, 同时重启多 CPU 系统。	○
7020	在多 CPU 系统里, CPU 故障出现在操作模式的“PLC 的停止错误使所有站停止”被选择的站点处。 (在有 CPU 故障出现的站以外的站点上的过程 CPU/高性能型 CPU 模块上检测到错误)。	使用外围设备读取错误的个别信息, 检查引起 CPU 模块故障的 CPU 的错误, 同时去除错误。	
9000	警报器设定为 0N。	使用外围设备读取错误的个别信息, 检查对应于数字值(警报器号码)的程序。	○
9010	错误通过 CHK 指令被检测到。	使用外围设备读取错误的个别信息, 检查对应于数字值(错误号码)的程序。	
9020	在自动写入标转 ROM 时, 到 ROM 的数据存储正常结束。 (BOOT LED 也闪烁)。	对标准 ROM 设定参数激活驱动, 再次接通电源, 从标准 ROM 进行引导操作。	○
10000	在多 CPU 系统, 错误出现在不是过程 CPU/高性能型 QCPU 的 CPU 模块里。	使用对应于 CPU 模块的软件包来检查错误出现的详细信息。	○

*1 在圆括号 () 里的字符表示存储个别信息的特殊寄存器的号码。

11.3 复位错误

Q/QnA 只对那些允许 CPU 继续运行的错误执行错误复位操作。

错误复位步骤如下指示：

- 1) 消除错误的原因。
- 2) 存储被复位的错误代码到特殊寄存器 SD50。
- 3) 打开特殊继电器 SM50。
- 4) 复位错误。

在复位错误后重启 CPU，与错误有关的特殊继电器，特殊寄存器，LED 和 LED 显示单元恢复到在错误出现前的状态。

在错误复位后，如果相同的错误再次出现，再次将错误登记到故障历史中。

如果多于一个警报器被检测到，错误复位操作只复位最先被检测到的 F 号码。

要点	
	错误代码的较低 1 位存储到 SD50，错误复位操作忽略。 (示例) <ul style="list-style-type: none">● 当错误代码为 2100 和 2111 的错误出现时，复位错误代码 2100 而不复位错误代码 2111。● 当错误代码为 2100 和 2101 的错误出现时，复位错误代码 2100 也复位错误代码 2101。

附录

附录 1 操作处理时间

1.1 定义

- (1) 在 Q/QnACPU 中，操作处理时间是下列时间的总和：
- 每个指令处理时间的总和
 - END 处理时间
 - I/O 刷新时间
- (2) 指令处理时间
这是列在附录 1.2 和 1.3 中的每条指令处理时间的总和
- (3) END 处理时间
END 处理时间是下列时间的总和：
- 在附录 1.2 和 1.3 中指出的 END 指令时间
 - MELSECNET 相关的刷新时间
 - 与外围设备间的通讯处理时间
 - 与如串行通讯模块等设备的通讯时间
- (4) I/O 刷新时间
$$\text{I/O 刷新时间} = (\text{输入点数}/16) * N1 + (\text{输出点数}/16) * N2$$

CPU 类型名称	N1 (μs)			N2 (μs)						
	Q 主基板	Q 扩展基板	QA 扩展基板	Q 主基板	Q 扩展基板	QA 扩展基板				
Q00JCPU	2. 5	3. 3	—	1. 3	2. 3	—				
Q00CPU	2. 4	3. 2	—	1. 3	2. 3	—				
Q01CPU	2. 3	3. 1	—	1. 3	2. 3	—				
Q02CPU	2. 2	2. 9	4. 3	1. 3	2. 1	3. 5				
Q02HCPU	1. 7	2. 4	3. 7	1. 3	2. 1	3. 5				
Q06HCPU			—			—				
Q12HCPU										
Q25HCPU			—			—				
Q12PHCPU	5. 2	4. 8	—	5. 0	4. 65	—				
Q25PHCPU										
Q2ASCPU (S1)	5. 2			5. 0						
Q2ACPU	4. 8			4. 65						
Q3ACPU	4. 8			4. 65						
Q2ASHCPU (S1)	4. 34			4. 26						
Q4ACPU	4. 34			4. 26						
Q4ARCPU	4. 34			4. 26						

1.2 基本型 QCPU 的操作处理时间

每条指令的处理时间列在下表中。
指令的处理时间会由于指令源和目标种类不同而产生相当大的变化，因此包含在下表中的值应作为处理时间的一套基本准则，而不能作为严格准确的时间来计算。

附录

(1) 顺控指令

指令	条件(软元件)	处理时间(μs)		
		Q00JCPU	Q00CPU	Q01CPU
LD LDI AND ANI OR ORI	X0	0.20	0.16	0.10
	D0, 0	0.30	0.24	0.15
LDP LDF ANDP ANDF ORP ORF	X0	0.30	0.24	0.15
	D0, 0			
ANB ORB MPS MRD MPP		0.20	0.16	0.10
INV	当不被执行时	0.20	0.16	0.10
	当被执行时			
MEP MEF	当不被执行时	0.30	0.24	0.15
	当被执行时			
EGP	当不改变时 (OFF → OFF) (ON → ON)	0.20	0.16	0.10
	当改变时 (OFF → ON) (ON → OFF)			
EGF	当不改变时 (OFF → OFF) (ON → ON)	17	9.5	9.4
	当改变时 (OFF → ON) (ON → OFF)	18	14	14

指令	条件(软元件)				处理时间(μs)				
					Q00JCPU	Q00CPU	Q01CPU		
OUT	当不改变时(OFF → OFF, ON → ON)				Y0	0.20	0.16	0.10	
					D0, 0	0.40	0.32	0.20	
	当改变时(OFF → ON, ON → OFF)				Y0	0.20	0.16	0.10	
					D0, 0	0.40	0.32	0.20	
	F	当为 OFF 时				24	20	19	
		当为 ON 时	当被显示时			260	210	200	
			显示完成时			205	165	155	
	T	当不被执行时				1.1	0.88	0.55	
		当被执行时	时间用完后			1.1	0.88	0.55	
			当增加时	K	1.1	0.88	0.55		
				D	1.2	0.96	0.60		
		C	当不被执行时				1.1	0.88	0.55
	当被执行时		时间用完后			1.1	0.88	0.55	
			当增加时	K	1.1	0.88	0.55		
				D	1.2	0.96	0.6		
OUTH	T		当不被执行时				1.1	0.88	0.55
		当被执行时	时间用完后			1.1	0.88	0.55	
			当增加时	K	1.1	0.88	0.55		
				D	1.2	0.96	0.6		
SET	Y	当不被执行时				0.20	0.16	0.10	
		当被执行时	当不发生改变时 (ON → ON)			0.20	0.16	0.10	
			当发生改变时 (OFF → ON)			0.20	0.16	0.10	
							0.20	0.16	0.10
	D0, 0	当不被执行时				0.40	0.32	0.20	
		当被执行时	当不发生改变时 (ON → ON)			0.40	0.32	0.20	
			当发生改变时 (OFF → ON)			0.40	0.32	0.20	
							0.40	0.32	0.20
F	当不被执行时				0.50	0.44	0.25		
	当被执行时	当被显示时			255	205	195		
		显示结束时			195	160	150		
						195	160	150	
RST	Y	当不被执行时				0.20	0.16	0.10	
		当被执行时	当不发生改变时 (OFF → OFF)			0.20	0.16	0.10	
			当发生改变时 (ON → OFF)			0.20	0.16	0.10	
							0.20	0.16	0.10
	D0, 0	当不被执行时				0.40	0.32	0.20	
		当被执行时	当不发生改变时 (ON → ON)			0.40	0.32	0.20	
			当发生改变时 (ON → OFF)			0.40	0.32	0.20	
							0.40	0.32	0.20
	SM	当不被执行时				0.20	0.16	0.10	
		当被执行时				0.20	0.16	0.10	
	F	当不被执行时				0.48	0.44	0.25	
		当被执行时	当被显示时			75	69	65	
			显示结束时			43	35	33	
							43	35	33
	T, C	D	当不被执行时				0.80	0.64	0.40
			当被执行时				1.0	0.80	0.50
		D	当不被执行时				0.4	0.32	0.20
			当被执行时				0.60	0.48	0.30
Z		当不被执行时				0.50	0.40	0.25	
		当被执行时				9.4	7.9	7.4	
R		当不被执行时				—	0.32	0.20	
		当被执行时				—	0.48	0.30	
PLS					12	9.5	9.2		
PLF					11	9.5	8.9		
FF	Y	当不被执行时				0.68	0.40	0.25	
当被执行时				7.5	6.2	5.7			
DELTA	DY0	当不被执行时				0.50	0.40	0.25	
当被执行时				26	21	21			
DELTAP	DY0	当不被执行时				0.48	0.40	0.25	
当被执行时				58	45	43			
SFT	当不被执行时				0.50	0.34	0.25		
SFTP	当被执行时				12	8.7	8.3		
MC	M0				0.40	0.32	0.20		
	D0, 0				3.3	2.9	2.8		
MCR					0.20	0.16	0.10		
	执行错误检查时				660	530	480		
FEND	不执行错误检查时								
END	● 电池检查				660	530	480		
	● 烧坏保险丝检查								
	● I/O 模块验证								
NOP					0.20	0.16	0.10		
NOPLF					0.20	0.16	0.10		
PAGE					0.20	0.16	0.10		

(2) 基本指令




































当指令不被执行时，处理时间按如下方式计算：

Q00JCPU 0.20 × (每条指令的步数 + 1) μs

Q00CPU 0.16 × (每条指令的步数 + 1) μs

Q01CPU 0.10 × (每条指令的步数 + 1) μs

指令	条件(软元件)		处理时间(μs)		
			Q00JCPU	Q00CPU	Q01CPU
LD =	当导通时		0.80	0.64	0.40
	当不导通时		0.80	0.64	0.40
AND =	当不被执行时		0.70	0.56	0.35
	当被执行时	当导通时	0.80	0.64	0.40
		当不导通时	0.80	0.64	0.40
	当不被执行时		0.70	0.56	0.35
OR =	当被执行时	当导通时	0.80	0.64	0.40
		当不导通时	0.80	0.64	0.40
	当不被执行时		0.70	0.56	0.35
	当不被执行时		0.70	0.56	0.35
LD <	当导通时		0.80	0.64	0.40
	当不导通时		0.80	0.64	0.40
AND <	当不被执行时		0.70	0.56	0.35
	当被执行时	当导通时	0.80	0.64	0.40
		当不导通时	0.80	0.64	0.40
	当不被执行时		0.70	0.56	0.35
OR <	当被执行时	当导通时	0.80	0.64	0.40
		当不导通时	0.80	0.64	0.40
	当不被执行时		0.70	0.56	0.35
	当不被执行时		0.70	0.56	0.35
LD >	当导通时		0.80	0.64	0.40
	当不导通时		0.80	0.64	0.40
AND >	当不被执行时		0.70	0.56	0.35
	当被执行时	当导通时	0.80	0.64	0.40
		当不导通时	0.80	0.64	0.40
	当不被执行时		0.70	0.56	0.35
OR >	当被执行时	当导通时	0.80	0.64	0.40
		当不导通时	0.80	0.64	0.40
	当不被执行时		0.70	0.56	0.35
	当不被执行时		0.70	0.56	0.35
LD < =	当导通时		0.80	0.64	0.40
	当不导通时		0.80	0.64	0.40
AND < =	当不被执行时		0.70	0.56	0.35
	当被执行时	当导通时	0.80	0.64	0.40
		当不导通时	0.80	0.64	0.40
	当不被执行时		0.70	0.56	0.35
OR < =	当被执行时	当导通时	0.80	0.64	0.40
		当不导通时	0.80	0.64	0.40
	当不被执行时		0.70	0.56	0.35
	当不被执行时		0.70	0.56	0.35
LD <	当导通时		0.80	0.64	0.40
	当不导通时		0.80	0.64	0.40
AND <	当不被执行时		0.70	0.56	0.35
	当被执行时	当导通时	0.80	0.64	0.40
		当不导通时	0.80	0.64	0.40
	当不被执行时		0.70	0.56	0.35
OR <	当被执行时	当导通时	0.80	0.64	0.40
		当不导通时	0.80	0.64	0.40
	当不被执行时		0.70	0.56	0.35
	当不被执行时		0.70	0.56	0.35
LD > =	当导通时		0.80	0.64	0.40
	当不导通时		0.80	0.64	0.40
AND > =	当不被执行时		0.70	0.56	0.35
	当被执行时	当导通时	0.80	0.64	0.40
		当不导通时	0.80	0.64	0.40
	当不被执行时		0.70	0.56	0.35
OR > =	当被执行时	当导通时	0.80	0.64	0.40
		当不导通时	0.80	0.64	0.40
	当不被执行时		0.70	0.56	0.35
	当不被执行时		0.70	0.56	0.35

指令	条件(软元件)	处理时间(μs)		
		Q00JCPU	Q00CPU	Q01CPU
LDD =	当导通时	1.0	0.80	0.50
	当不导通时	1.0	0.80	0.50
ANDD =	当不执行时	0.80	0.64	0.40
	当被执行时	当导通时	1.0	0.80
		当不导通时	1.0	0.50
ORD =	当不执行时	0.80	0.64	0.40
	当被执行时	当导通时	1.0	0.80
		当不导通时	1.0	0.50
LDD < >	当导通时	1.0	0.80	0.50
	当不导通时	1.0	0.80	0.50
ANDD < >	当不执行时	0.80	0.64	0.40
	当被执行时	当导通时	1.0	0.80
		当不导通时	1.0	0.50
ORD < >	当不执行时	0.80	0.64	0.40
	当被执行时	当导通时	1.0	0.80
		当不导通时	1.0	0.50
LDD >	当导通时	1.0	0.80	0.50
	当不导通时	1.0	0.80	0.50
ANDD >	当不执行时	0.80	0.64	0.40
	当被执行时	当导通时	1.0	0.80
		当不导通时	1.0	0.50
ORD >	当不执行时	0.80	0.64	0.40
	当被执行时	当导通时	1.0	0.80
		当不导通时	1.0	0.50
LDD < =	当导通时	1.0	0.80	0.50
	当不导通时	1.0	0.80	0.50
ANDD < =	当不执行时	0.80	0.64	0.40
	当被执行时	当导通时	1.0	0.80
		当不导通时	1.0	0.50
ORD < =	当不执行时	0.80	0.64	0.40
	当被执行时	当导通时	1.0	0.80
		当不导通时	1.0	0.50
LDD <	当导通时	1.0	0.80	0.50
	当不导通时	1.0	0.80	0.50
ANDD <	当不执行时	0.80	0.64	0.40
	当被执行时	当导通时	1.0	0.80
		当不导通时	1.0	0.50
ORD <	当不执行时	0.80	0.64	0.40
	当被执行时	当导通时	1.0	0.80
		当不导通时	1.0	0.50
LDD > =	当导通时	1.0	0.80	0.50
	当不导通时	1.0	0.80	0.50
ANDD > =	当不执行时	0.80	0.64	0.40
	当被执行时	当导通时	1.0	0.80
		当不导通时	1.0	0.50
ORD > =	当不执行时	0.80	0.64	0.40
	当被执行时	当导通时	1.0	0.80
		当不导通时	1.0	0.50
BKCMPI =    n	n = 1	130	105	97
BKCMPI = P    n	n = 96	205	175	165
BKCMPI < >    n	n = 1	130	105	98
BKCMPI < > P    n	n = 96	210	180	165
BKCMPI >    n	n = 1	130	105	97
BKCMPI > P    n	n = 96	210	180	165
BKCMPI > =    n	n = 1	130	105	98
BKCMPI > = P    n	n = 96	205	175	165
BKCMPI <    n	n = 1	130	105	98
BKCMPI < P    n	n = 96	210	180	165
BKCMPI < =    n	n = 1	130	105	97
BKCMPI < = P   n	n = 96	205	175	165

指令	条件(软元件)	处理时间(μs)		
		Q00JCPU	Q00CPU	Q01CPU
+ $\text{S} \text{D}$ + P $\text{S} \text{D}$	当导通时	1.0	0.80	0.50
+ $\text{S1} \text{S2} \text{D}$ + P $\text{S1} \text{S2} \text{D}$	当导通时	1.2	0.96	0.60
- $\text{S} \text{D}$ - P $\text{S} \text{D}$	当导通时	1.0	0.80	0.50
- $\text{S1} \text{S2} \text{D}$ - P $\text{S1} \text{S2} \text{D}$	当导通时	1.2	0.96	0.60
D + $\text{S} \text{D}$ D + P $\text{S} \text{D}$	当导通时	1.3	1.04	0.65
D + $\text{S1} \text{S2} \text{D}$ D + P $\text{S1} \text{S2} \text{D}$	当导通时	1.5	1.2	0.75
D - $\text{S} \text{D}$ D - P $\text{S} \text{D}$	当导通时	1.3	1.04	0.65
D - $\text{S1} \text{S2} \text{D}$ D - P $\text{S1} \text{S2} \text{D}$	当导通时	1.5	1.2	0.75
* $\text{S1} \text{S2} \text{D}$ * P $\text{S1} \text{S2} \text{D}$	当导通时	1.1	0.88	0.55
/ $\text{S1} \text{S2} \text{D}$ /P $\text{S1} \text{S2} \text{D}$		19	16	15
D * $\text{S1} \text{S2} \text{D}$ D * P $\text{S1} \text{S2} \text{D}$		41	34	31
D / $\text{S1} \text{S2} \text{D}$ D / P $\text{S1} \text{S2} \text{D}$		28	23	21
B + $\text{S} \text{D}$ B + P $\text{S} \text{D}$		34	28	26
B + $\text{S1} \text{S2} \text{D}$ B + P $\text{S1} \text{S2} \text{D}$		47	39	37
B - $\text{S} \text{D}$ B - P $\text{S} \text{D}$		34	28	26
B - $\text{S1} \text{S2} \text{D}$ B - P $\text{S1} \text{S2} \text{D}$		48	40	38
DB + $\text{S} \text{D}$ DB + P $\text{S} \text{D}$		58	48	44
DB + $\text{S1} \text{S2} \text{D}$ DB + P $\text{S1} \text{S2} \text{D}$		60	49	46
DB - $\text{S} \text{D}$ DB - P $\text{S} \text{D}$		59	48	45
DB - $\text{S1} \text{S2} \text{D}$ DB - P $\text{S1} \text{S2} \text{D}$		60	51	45
B * $\text{S1} \text{S2} \text{D}$ B * P $\text{S1} \text{S2} \text{D}$		42	35	33
B / $\text{S1} \text{S2} \text{D}$ B / P $\text{S1} \text{S2} \text{D}$		48	40	37
DB * $\text{S1} \text{S2} \text{D}$ DB * P $\text{S1} \text{S2} \text{D}$		140	120	110
DB / $\text{S1} \text{S2} \text{D}$ DB / P $\text{S1} \text{S2} \text{D}$		83	69	65
BK + $\text{S1} \text{S2} \text{D} \text{ n}$ BK + P $\text{S1} \text{S2} \text{D} \text{ n}$	n = 1 n = 96	105 185	86 155	80 140
BK - $\text{S1} \text{S2} \text{D} \text{ n}$ BK - P $\text{S1} \text{S2} \text{D} \text{ n}$	n = 1 n = 96	105 185	86 155	80 140

指令	条件(软元件)	处理时间(μs)		
		Q00JCPU	Q00CPU	Q01CPU
INC INCP		0.70	0.56	0.35
DINC DINCP		0.90	0.72	0.45
DEC DECP		0.70	0.56	0.35
DDEC DDECP		0.90	0.72	0.45
BCD BCDP		20	16	15
DBCD DBCDP		26	21	20
BIN BINP		19	16	15
DBIN DBINP		22	18	17
DBL DBLP		19	16	15
WORD WORDP		23	19	17
GRY GRYP		19	16	15
DGRY DGRYP		23	19	17
GBIN GBINP		52	42	40
DGBIN DGBINP		110	88	84
NEG NEGP		16	13	12
DNEG DNEGP		19	17	15

指令	条件(软元件)	处理时间(μs)		
		Q00JCPU	Q00CPU	Q01CPU
BKBCD \textcircled{S} \textcircled{D} n	n = 1	78	63	57
BKBCDP \textcircled{S} \textcircled{D} n	n = 96	315	275	250
BKBIN \textcircled{S} \textcircled{D} n	n = 1	74	61	57
BKBINP \textcircled{S} \textcircled{D} n	n = 96	285	255	230
MOV	\textcircled{S} = D0, \textcircled{D} = D1	0.70	0.56	0.35
MOVP	\textcircled{S} = D0, \textcircled{D} = J1\W1	155	130	120
DMOV	\textcircled{S} = D0, \textcircled{D} = D1	0.90	0.72	0.45
DMOVP	\textcircled{S} = D0, \textcircled{D} = J1\W1	165	135	120
\$ MOV		46	38	35
\$ MOVP		98	80	73
CML		0.70	0.56	0.35
CMLP				
DCML		0.90	0.72	0.45
DCMLP				
BMOV \textcircled{S} \textcircled{D} n	n = 1	27	21	20
BMOVP \textcircled{S} \textcircled{D} n	n = 96	72	62	53
FMOV \textcircled{S} \textcircled{D} n	n = 1	23	19	17
FMOVP \textcircled{S} \textcircled{D} n	n = 96	48	41	36
XCH		7.6	6.3	5.7
XCHP				
DXCH		9.5	8.0	7.1
DXCHP				
BXCH \textcircled{S} \textcircled{D} n	n = 1	62	51	48
BXCHP \textcircled{S} \textcircled{D} n	n = 96	165	140	125
SWAP		17	14	13
SWAPP				
CJ		10	8.5	8.1
SCJ		10	8.5	8.1
JMP		11	8.5	8.1
GOEND		3.3	2.9	2.8
DI		13	12	11
EI		14	11	11
IMASK		41	34	35
IRET		205	170	155
RFS RFSP	X n = 1	55	46	43
	X n = 96	79	64	59
	Y n = 1	54	45	41
	Y n = 96	73	61	56

(3) 应用指令

当指令不执行时，处理时间按如下方式计算：

Q00JCPU $0.20 \times (\text{每条指令的步数} + 1) \mu\text{s}$

Q00CPU $0.16 \times (\text{每条指令的步数} + 1) \mu\text{s}$

Q01CPU $0.10 \times (\text{每条指令的步数} + 1) \mu\text{s}$

指令	条件(软元件)	处理时间(μs)		
		Q00JCPU	Q00CPU	Q01CPU
WAND $\text{S} \text{ D}$ WANDP $\text{S} \text{ D}$	当导通时	1.0	0.80	0.50
WAND $\text{S1} \text{ S2} \text{ D}$ WANDP $\text{S1} \text{ S2} \text{ D}$	当导通时	1.2	0.96	0.60
DAND $\text{S} \text{ D}$ DANDP $\text{S} \text{ D}$	当导通时	1.3	1.04	0.65
DAND $\text{S1} \text{ S2} \text{ D}$ DANDP $\text{S1} \text{ S2} \text{ D}$	当导通时	1.5	1.2	0.75
BKAND $\text{S1} \text{ S2} \text{ D} \text{ n}$ BKANDP $\text{S1} \text{ S2} \text{ D} \text{ n}$	$n = 1$ $n = 96$	110 185	87 155	79 140
WOR $\text{S} \text{ D}$ WORP $\text{S} \text{ D}$	当导通时	1.0	0.80	0.50
WOR $\text{S1} \text{ S2} \text{ D}$ WORP $\text{S1} \text{ S2} \text{ D}$	当导通时	1.2	0.96	0.60
DOR $\text{S} \text{ D}$ DORP $\text{S} \text{ D}$	当导通时	1.3	1.04	0.65
DOR $\text{S1} \text{ S2} \text{ D}$ DORP $\text{S1} \text{ S2} \text{ D}$	当导通时	1.5	1.2	0.75
BKOR $\text{S1} \text{ S2} \text{ D} \text{ n}$ BKORP $\text{S1} \text{ S2} \text{ D} \text{ n}$	$n = 1$ $n = 96$	110 185	87 155	81 140
WXOR $\text{S} \text{ D}$ WXORP $\text{S} \text{ D}$	当导通时	1.0	0.80	0.50
WXOR $\text{S1} \text{ S2} \text{ D}$ WXORP $\text{S1} \text{ S2} \text{ D}$	当导通时	1.2	0.96	0.60
DXOR $\text{S} \text{ D}$ DXORP $\text{S} \text{ D}$	当导通时	1.3	1.04	0.65
DXOR $\text{S1} \text{ S2} \text{ D}$ DXORP $\text{S1} \text{ S2} \text{ D}$	当导通时	1.5	1.2	0.75
BKXOR $\text{S1} \text{ S2} \text{ D} \text{ n}$ BKXORP $\text{S1} \text{ S2} \text{ D} \text{ n}$	$n = 1$ $n = 96$	110 183	87 155	81 140
WXNR $\text{S} \text{ D}$ WXNRP $\text{S} \text{ D}$	当导通时	1.0	0.80	0.50
WXNR $\text{S1} \text{ S2} \text{ D}$ WXNRP $\text{S1} \text{ S2} \text{ D}$	当导通时	1.2	0.96	0.60
DNXR $\text{S} \text{ D}$ DNXRP $\text{S} \text{ D}$	当导通时	1.3	1.04	0.65
DNXR $\text{S1} \text{ S2} \text{ D}$ DNXRP $\text{S1} \text{ S2} \text{ D}$	当导通时	1.5	1.2	0.75
BKXNR $\text{S1} \text{ S2} \text{ D} \text{ n}$ BKXNRP $\text{S1} \text{ S2} \text{ D} \text{ n}$	$n = 1$ $n = 96$	110 185	87 155	82 140
ROR $\text{D} \text{ n}$ RORP $\text{D} \text{ n}$	$n = 1$ $n = 15$	13 13	11 11	9.7 9.7
RCR $\text{D} \text{ n}$ RCRP $\text{D} \text{ n}$	$n = 1$ $n = 15$	15 15	12 13	12 12
ROL $\text{D} \text{ n}$ ROLP $\text{D} \text{ n}$	$n = 1$ $n = 15$	13 13	11 11	10 10
RCL $\text{D} \text{ n}$ RCLP $\text{D} \text{ n}$	$n = 1$ $n = 15$	15 16	13 13	12 12

指令	条件(软元件)	处理时间(μs)		
		Q00JCPU	Q00CPU	Q01CPU
DROR \textcircled{D} n	n = 1	15	12	12
DRORP \textcircled{D} n	n = 31	15	13	12
DRCR \textcircled{D} n	n = 1	17	14	14
DRCRP \textcircled{D} n	n = 31	18	16	15
DROL \textcircled{D} n	n = 1	14	13	12
DROL P \textcircled{D} n	n = 31	14	13	12
DRCL \textcircled{D} n	n = 1	18	15	14
DRCLP \textcircled{D} n	n = 31	20	17	16
SFR \textcircled{D} n	n = 1	13	10	9.7
SFRP \textcircled{D} n	n = 15	13	11	9.5
SFL \textcircled{D} n	n = 1	12	10	9.5
SFLP \textcircled{D} n	n = 15	12	9.8	9.5
BSFLR \textcircled{D} n	n = 1	42	35	33
BSFLRP \textcircled{D} n	n = 96	69	58	54
BSFL \textcircled{D} n	n = 1	41	34	32
BSFLP \textcircled{D} n	n = 96	63	53	50
DSFR \textcircled{D} n	n = 1	19	16	15
DSFRP \textcircled{D} n	n = 96	71	61	53
DSFL \textcircled{D} n	n = 1	19	16	15
DSFLP \textcircled{D} n	n = 96	70	60	52
BSET \textcircled{D} n	n = 1	27	22	20
BSETP \textcircled{D} n	n = 15	27	22	20
BRST \textcircled{D} n	n = 1	27	22	21
BRSTP \textcircled{D} n	n = 15	27	22	21
TEST $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D}		35	30	27
TESTP $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D}				
DTEST $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D}		37	31	28
DTESTP $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D}				
BKRST \textcircled{S} n	n = 1	49	41	38
BKRSTP \textcircled{S} n	n = 96	64	54	50
SER $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D} n SERP $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D} n	n = 1	所有的都匹配	56	54
		一个都不匹配	56	54
	n = 96	所有的都匹配	280	240
		一个都不匹配	280	240
DSER $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D} n DSERP $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D} n	n = 1	所有的都匹配	71	67
		一个都不匹配	71	67
	n = 96	所有的都匹配	495	415
		一个都不匹配	500	415
SUM	\textcircled{S} = 0	32	26	25
SUMP	\textcircled{S} = FFFF	27	22	21
DSUM	\textcircled{S} = 0	54	44	42
DSUMP	\textcircled{S} = FFFFFFFF ₁₆	54	44	42
DECO \textcircled{S} \textcircled{D} n	n = 2	60	50	46
DECOP \textcircled{S} \textcircled{D} n	n = 8	80	65	61
ENCO \textcircled{S} \textcircled{D} n ENCOP \textcircled{S} \textcircled{D} n	n = 2	M1 = ON	66	55
		M4 = ON	66	54
	n = 8	M1 = ON	90	76
		M256 = ON	76	74
SEG		8.0	6.8	6.1
SEGP				
DIS \textcircled{S} \textcircled{D} n	n = 1	47	39	36
DISP \textcircled{S} \textcircled{D} n	n = 4	53	43	40
UNI \textcircled{S} \textcircled{D} n	n = 1	54	44	41
UNIP \textcircled{S} \textcircled{D} n	n = 4	60	49	46
NDIS $\textcircled{S1}$ \textcircled{D} $\textcircled{S2}$		92	76	38
NDISP $\textcircled{S1}$ \textcircled{D} $\textcircled{S2}$				
NUNI $\textcircled{S1}$ \textcircled{D} $\textcircled{S2}$		47	39	36
NUNIP $\textcircled{S1}$ \textcircled{D} $\textcircled{S2}$				
WTOB \textcircled{S} \textcircled{D} n	n = 1	56	46	42
WTOBP \textcircled{S} \textcircled{D} n	n = 96	190	155	145
BTOW \textcircled{S} \textcircled{D} n	n = 1	56	46	42
BTOWP \textcircled{S} \textcircled{D} n	n = 96	190	155	145

指令	条件(软元件)	处理时间(μs)		
		Q00JCPU	Q00CPU	Q01CPU
MAX \textcircled{S} \textcircled{D} n	n = 1	48	40	36
MAXP \textcircled{S} \textcircled{D} n	n = 96	300	240	235
MIN \textcircled{S} \textcircled{D} n	n = 1	48	40	36
MINP \textcircled{S} \textcircled{D} n	n = 96	300	240	235
DMAX \textcircled{S} \textcircled{D} n	n = 1	52	43	39
DMAXP \textcircled{S} \textcircled{D} n	n = 96	600	490	460
DMIN \textcircled{S} \textcircled{D} n	n = 1	52	43	39
DMINP \textcircled{S} \textcircled{D} n	n = 96	585	475	445
SORT $\textcircled{S1}$ n $\textcircled{S2}$ $\textcircled{D1}$ $\textcircled{D2}$	n = 1	66	55	50
	n = 96	105	86	80
DSORT $\textcircled{S1}$ n $\textcircled{S2}$ $\textcircled{D1}$ $\textcircled{D2}$	n = 1	98	57	52
	n = 96	115	96	88
WSUM \textcircled{S} \textcircled{D} n	n = 1	52	43	40
WSUMP \textcircled{S} \textcircled{D} n	n = 96	175	140	135
DWSUM \textcircled{S} \textcircled{D} n	n = 1	61	51	46
DWSUMP \textcircled{S} \textcircled{D} n	n = 96	515	420	395
FOR n	n = 0	11	8.9	8.1
NEXT		8.8	7.3	6.8
BREAK		37	30	28
BREAKP				
CALL Pn		17	14	13
CALLP Pn				
CALL Pn $\textcircled{S1}$ 到 $\textcircled{S5}$		245	200	190
CALLP Pn $\textcircled{S1}$ 到 $\textcircled{S5}$				
RET	返回正常程序	16	13	12
FCALL Pn		29	24	22
FCALLP Pn				
FCALL Pn $\textcircled{S1}$ 到 $\textcircled{S5}$		250	205	190
FCALLP Pn $\textcircled{S1}$ 到 $\textcircled{S5}$				

指令	条件(软元件)	处理时间(μs)		
		Q00JCPU	Q00CPU	Q01CPU
COM		110	77	72
IX		65	54	51
IXEND		30	26	25
IXDEV	触点数为 1	145	120	110
+ IXSET	触点数为 14	770	630	585
FIFW	数据点数为 0	36	32	28
FIFWP	数据点数为 96	36	32	28
FIFR	数据点数为 1	45	41	36
FIFRP	数据点数为 96	93	82	70
FPOP	数据点数为 1	40	37	32
FPOPP	数据点数为 96	40	37	32
FINS	数据点数为 0	53	44	38
FINSP	数据点数为 96	100	89	76
FDEL	数据点数为 1	60	50	43
FDLPP	数据点数为 96	110	95	82
FROM n1 n2 \oplus n3	n3 = 1	125	105	93
FROMP n1 n2 \oplus n3	n3 = 1000	740	695	685
*				
DFRO n1 n2 \oplus n3	n3 = 1	130	110	100
DFROP n1 n2 \oplus n3	n3 = 500	745	695	675
*				
TO n1 n2 \oplus n3	n3 = 1	120	105	92
TOP n1 n2 \oplus n3	n3 = 1000	735	680	645
*				
DTO n1 n2 \oplus n3	n3 = 1	130	110	99
DTOP n1 n2 \oplus n3	n3 = 500	740	680	640
*				

备注

- 1) *: FROM/TO 指令的处理时间随插槽数和安装的模块数而变化。
(CPU 的处理时间也随扩展基板类型而变化。)

指令	条件(软元件)	处理时间(μs)		
		Q00JCPU	Q00CPU	Q01CPU
LIMIT LIMITP		34	28	26
DLIMIT DLIMITP		41	34	30
BAND BANDP		33	28	25
DBAND DBANDP		40	34	30
ZONE ZONEP		31	25	24
DZONE DZONEP		37	29	28
RSET RSETP		—	18	16
DATERD DATERDP		30	25	23
DATEWR DATEWRP		69	57	54
DATE + DATE + P	无位数增加	47	39	36
	位数增加	50	42	38
DATE - DATE - P	无位数增加	47	40	36
	位数增加	50	42	38
SECOND SECONDP		28	24	22
HOUR HOURP		38	32	29
WDT WDTP		18	15	14
DUTY		41	36	32
ZRRDB ZRRDBP		—	24	22
ZRWRB ZRWRBP		—	27	24
ADRSET ADRSETP		23	19	18
ZPUSH ZPUSHP		38	33	30
ZPOP ZPOPP		37	31	29
ZCOM		105	82	80

(4) QCPU 指令的处理时间(只包括 QCPU 指令)

(a) 功能版本 A 中可用的指令

指令	条件(软元件)	处理时间(μs)		
		Q00JCPU	Q00CPU	Q01CPU
UNIRD	n = 1	96	80	74
UNIRDP	n = 16	440	370	340

(4) 可以被头五位系列号为“04122”或更大数字的产品执行的指令

指令	条件(软元件)		处理时间(μs)		
			Q00JCPU	Q00CPU	Q01CPU
LDE =	单精度	当导通时	43.0	35.5	33.0
		当不导通时	46.0	38.0	35.5
ANDE =	单精度	当不被执行时	1.5	1.2	1.0
		当被执行时	当导通时	35.5	29.5
			当不导通时	42.0	35.0
ORE =	单精度	当不被执行时	1.5	1.2	1.0
		当被执行时	当导通时	42.0	35.0
			当不导通时	37.0	31.0
LDE < >	单精度	当导通时	46.0	38.0	35.5
		当不导通时	43.5	36.0	33.0
ANDE < >	单精度	当不被执行时	1.5	1.2	1.0
		当被执行时	当导通时	38.5	31.5
			当不导通时	39.5	33.0
ORE < >	单精度	当不被执行时	1.5	1.2	1.0
		当被执行时	当导通时	45.0	37.5
			当不导通时	34.5	29.0
LDE >	单精度	当导通时	46.0	37.5	35.5
		当不导通时	46.0	38.5	35.0
ANDE >	单精度	当不被执行时	1.5	1.2	1.0
		当被执行时	当导通时	38.5	32.0
			当不导通时	42.0	35.0
ORE >	单精度	当不被执行时	1.5	1.2	1.0
		当被执行时	当导通时	45.0	37.5
			当不导通时	37.0	31.0

指令	条件(软元件)		处理时间(μs)			
			Q00JCPU	Q00CPU	Q01CPU	
LDE < =	单精度	当导通时		45.5	37.5	35.0
		当不导通时		46.5	38.5	35.5
ANDE < =	单精度	当不被执行时		1.5	1.2	1.0
		当被执行时	当导通时	38.5	31.5	29.0
			当不导通时	42.5	35.5	32.5
ORE < =	单精度	当不被执行时		1.5	1.2	1.0
		当被执行时	当导通时	45.0	37.5	34.5
			当不导通时	37.5	31.5	28.5
LDE <	单精度	当导通时		45.5	37.5	35.0
		当不导通时		46.5	38.5	35.5
ANDE <	单精度	当不被执行时		1.5	1.2	1.0
		当被执行时	当导通时	38.0	31.5	29.0
			当不导通时	42.5	35.5	32.5
ORE <	单精度	当不被执行时		1.5	1.2	1.0
		当被执行时	当导通时	45.0	37.5	34.5
			当不导通时	37.5	31.5	29.0
LDE > =	单精度	当导通时		45.5	38.0	35.5
		当不导通时		46.5	38.0	35.0
ANDE > =	单精度	当不被执行时		1.5	1.2	1.0
		当被执行时	当导通时	38.5	32.0	29.0
			当不导通时	42.5	35.5	32.5
ORE > =	单精度	当不被执行时		1.5	1.2	1.0
		当被执行时	当导通时	45.0	38.5	34.5
			当不导通时	37.5	31.0	28.5

指令	条件(软元件)		处理时间(μs)		
			Q00JCPU	Q00CPU	Q01CPU
E + \textcircled{S} \textcircled{D} E + P \textcircled{S} \textcircled{D}	单精度	\textcircled{S} = 0, \textcircled{D} = 0	29.5	25.0	23.0
		\textcircled{S} = 2^{127} , \textcircled{D} = 2^{127}	65.5	60.5	49.5
E + $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D} E + P $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D}	单精度	\textcircled{S} = 0, \textcircled{D} = 0	31.0	27.0	24.0
		\textcircled{S} = 2^{127} , \textcircled{D} = 2^{127}	66.5	56.0	51.0
E - \textcircled{S} \textcircled{D} E - P \textcircled{S} \textcircled{D}	单精度	\textcircled{S} = 0, \textcircled{D} = 0	29.5	25.0	23.0
		\textcircled{S} = 2^{127} , \textcircled{D} = 2^{127}	48.5	41.0	37.5
E - $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D} E - P $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D}	单精度	$\textcircled{S1}$ = 0, $\textcircled{S2}$ = 0	31.0	27.0	24.0
		$\textcircled{S1}$ = 2^{127} , $\textcircled{S2}$ = 2^{127}	50.5	42.5	38.5
E * $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D} E * P $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D}	单精度	$\textcircled{S1}$ = 0, $\textcircled{S2}$ = 0	30.0	25.5	23.0
		$\textcircled{S1}$ = 2^{126} , $\textcircled{S2}$ = 2^{127}	65.5	55.0	49.5
E / $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D} E / P $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D}	单精度	$\textcircled{S1}$ = 0, $\textcircled{S2}$ = 1	30.0	26.0	23.0
		$\textcircled{S1}$ = 2^{127} , $\textcircled{S2}$ = - 2^{126}	69.5	57.5	53.0
INT INTP	单精度	\textcircled{S} = 0	21.5	18.5	16.0
		\textcircled{S} = 32766.5	38.0	32.0	29.5
DINT DINTP	单精度	\textcircled{S} = 0	23.0	19.5	17.5
		\textcircled{S} = 1234567890.3	42.0	35.5	32.0
FLT FLTP	单精度	\textcircled{S} = 0	22.5	19.5	17.0
		\textcircled{S} = 7FFFFH	26.5	23.0	20.0
DFLT DFLTP	单精度	\textcircled{S} = 0	23.0	20.0	17.5
		\textcircled{S} = 7FFFFFFFH	26.0	23.5	19.5

指令	条件(软元件)		处理时间(μs)		
			Q00ICPU	Q00CPU	Q01CPU
ENEG	S = 0		20.5	17.0	15.5
ENEGP	S = E - 1.0		31.5	26.0	24.0
EMOV			1.5	1.2	1.0
EMOVPP					
ESTR			604.0	686.0	831.0
ESTRP					
EVAL	小数点格式 所有的 2 位		138.0	148.0	196.0
EVALP	指数格式 所有的 6 位		164.0	177.0	214.0
SIN	单精度		204.0	173.0	157.0
SNIP					
COS	单精度		187.0	158.0	144.0
COSP					
TAN	单精度		224.0	190.0	173.0
TANP					
RAD	单精度		51.0	43.0	39.0
RADP					
DEG	单精度		51.0	43.0	39.0
DEGP					
SQR	单精度		60.0	51.0	46.5
SQRP					
EXP	单精度	S = - 10	306.0	259.0	235.0
		S = 1	306.0	259.0	235.0
LOG	单精度	S = 1	73.0	61.5	56.0
		S = 10	301.0	255.0	232.0
LOGP					
RND			12.5	11.0	10.0
RNDP					
SRND			13.5	12.0	11.0
SRNDP					
COM	带 CPU 共享存储区的自动刷新	刷新范围: 2k 字 (0.5k 字被平均分配到所有站点)	—	920	880
	不带 CPU 共享存储区的自动刷新	—	—	150	135
FROM	从主站点的 CPU 共享存储区读	n3 = 1	—	100	90
		n3 = 320	—	440	420
	从其它站点的 CPU 共享存储区读	n3 = 1	—	110	105
		n3 = 320	—	305	290
TO	向主站点的 CPU 共享存储区写	n3 = 1	—	100	95
		n3 = 320	—	440	425
S. TO	向主站点的 CPU 共享存储区写	n3 = 1	—	205	195
		n3 = 320	—	545	525

1.3 高性能型 QCPU/过程控制 CPU/QnACPU 的操作处理时间

每条指令的处理时间列在下表中：
指令的处理时间会由于指令源和目标种类不同而产生相当大的变化，因此包含在下表中的值应作为处理时间的一套基本准则，而不能作为准确的时间来计算。

(1) 顺控指令

指令	条件(软元件)	处理时间(μs)						
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
LD LDI AND ANI OR ORI		0.20	0.15	0.075		0.079	0.034	0.034
LDP LDF ANDP ANDF ORP ORF		6.6	5.0	2.5		0.158	0.068	0.068
ANB ORB MPS MRD MPP		0.20	0.15	0.075		0.079	0.034	0.034
INV	当不被执行时	2.4	1.8	0.9		0.079	0.034	0.034
	当被执行时							
MEP MEF	当不被执行时	2.0	1.5	0.75		0.173	0.073	0.073
	当被执行时							
EGP EGF	当不发生改变时	0.6	0.3	0.15		0.158	0.068	0.068
	当发生改变时							

指令	条件(软元件)			处理时间(μs)							
				Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH	
OUT				0.40	0.30	0.15	0.158	0.068	0.068		
				0.40	0.30	0.15	0.158	0.068	0.068		
	F	当为 OFF 时		7.0	5.3	2.7	2.8	1.2	1.2		
		当为 ON 时	当被显示时	167	126	63	162	69.7	69.7		
	显示完成时		166	125	62	126	54	54			
	T	当不被执行时			1.6	1.2	0.6	0.63	0.27	0.27	
		当被执行时	时间用完后	1.6	1.2	0.6	0.63	0.27	0.27		
			当被增加时	K	1.6	1.2	0.6	0.63	0.27	0.27	
				D	1.6	1.2	0.6	0.63	0.27	0.27	
	C	当不被执行时			1.6	1.2	0.6	0.63	0.27	0.27	
		当被执行时	时间用完后	1.6	1.2	0.6	0.63	0.27	0.27		
			当被增加时	K	1.6	1.2	0.6	0.63	0.27	0.27	
				D	1.6	1.2	0.6	0.63	0.27	0.27	
OUTH	T	当不被执行时			1.6	1.2	0.6	0.63	0.27	0.27	
		当被执行时	时间用完后	1.6	1.2	0.6	0.63	0.27	0.27		
			当被增加时	K	1.6	1.2	0.6	0.63	0.27	0.27	
				D	1.6	1.2	0.15	0.63	0.27	0.27	
SET	当不被执行时			0.40	0.30	0.15	0.158	0.068	0.068		
	当被执行时		当不发生改变时(ON ON)	0.40	0.30	0.15	0.158	0.068	0.068		
			当发生改变时(OFF ON)	0.40	0.30	0.45	0.158	0.068	0.068		
	F	当不被执行时			1.2	0.90	104	0.47	0.20	0.20	
		当被执行时	当被显示时	277	208	0.45	161	69	69		
显示完成时			1.2	0.90	0.15	0.47	0.20	0.20			
RST	当不被执行时			0.40	0.30	0.15	0.158	0.068	0.068		
	当被执行时		当不发生改变时(OFF OFF)	0.40	0.30	0.15	0.158	0.068	0.068		
			当发生改变时(ON OFF)	0.40	0.30	0.15	0.158	0.068	0.068		
	SM	当不被执行时			0.40	0.30	0.15	0.158	0.068	0.068	
		当被执行时			0.40	0.30	0.15	0.158	0.068	0.068	
	F	当不被执行时			1.2	0.90	0.45	0.47	0.20	0.20	
		当被执行时	当被显示时	148	112	56	90	38	38		
			显示完成时	1.2	0.90	0.45	0.47	0.20	0.20		
	T, C	当不被执行时			1.4	1.1	0.6	0.63	0.27	0.27	
		当被执行时						0.63	0.27	0.27	
		D	当不被执行时			0.60	0.45	0.23	0.24	0.10	0.10
			当被执行时						0.24	0.10	0.10
	Z	当不被执行时			1.2	0.90	0.45	0.47	0.20	0.20	
		当被执行时			10.8	8.1	4.1	4.3	1.9	1.9	
		R	当不被执行时			1.0	0.75	0.38	0.40	0.17	0.17
当被执行时						0.40	0.17	0.17			
PLS PLF				2.6	2.0	0.98	1.0	0.44	0.44		
FF	Y	当不被执行时			1.2	0.90	0.45	0.47	0.20	0.20	
当被执行时						0.47	0.20	0.20			
DELTA DELTAP	DY0	当不被执行时			1.2	0.90	0.45	0.47	0.20	0.20	
当被执行时			16.8	14.1	11.1	5.9	2.6	2.6			

指令	条件(软元件)	处理时间(μs)						
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
SFT	当不被执行时	1.2	0.90	0.45		0.47	0.20	0.20
SFTP	当被执行时	4.2	3.2	1.6		1.66	0.71	0.71
MC		0.60	0.45	0.23		0.24	0.10	0.10
MCR		0.20	0.15	0.075		0.079	0.034	0.034
FEND END	执行错误检查时	1643	1236	618		348	150	150
	不执行错误检查时							
	<ul style="list-style-type: none"> ● 电池检查 ● 烧坏保险丝检查 ● I/O 模块验证 	1106	832	416		359	150	150
NOP		0.2	0.15	0.075		0.079	0.034	0.034
NOPLF PAGE		0.2	0.15	0.075		0.079	0.034	0.034

(2) 基本指令

当指令不被执行时，处理时间按如下方式计算：

Q2ACPU (S1), Q2ASCPU (S1) $0.20 \times (\text{每条指令的执行步数} + 1) \mu\text{s}$
 Q3ACPU $0.15 \times (\text{每条指令的执行步数} + 1) \mu\text{s}$
 Q2ASHCPU (S1), Q4ACPU, Q4ARCPU $0.075 \times (\text{每条指令的执行步数} + 1) \mu\text{s}$
 Q02CPU $0.079 \times (\text{每条指令的执行步数} + 1) \mu\text{s}$
 Q02CPU, Q06CPU, Q12HCPU, Q25HCPU
 Q12PHCPU, Q25PHCPU $0.034 \times (\text{每条指令的执行步数} + 1) \mu\text{s}$

指令	条件(软元件)	处理时间(μs)						
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
LD =	当导通时	3.8	2.9	1.5		0.24	0.10	0.10
	当不导通时	3.6	2.7	1.4		0.24	0.10	0.10
AND =	当不被执行时	1.4	1.1	0.55		0.24	0.10	0.10
	当被执行时	当导通时	2.8	2.1	1.1	0.24	0.10	0.10
		当不导通时	3.2	2.4	1.2	0.24	0.10	0.10
OR =	当不被执行时	1.4	1.1	0.55		0.24	0.10	0.10
	当被执行时	当导通时	3.8	2.9	1.5	0.24	0.10	0.10
		当不导通时	2.8	2.1	1.1	0.24	0.10	0.10
LD < >	当导通时	4.4	3.3	1.7		0.24	0.10	0.10
	当不导通时	3.6	2.7	1.4		0.24	0.10	0.10
AND < >	当不被执行时	1.4	1.1	0.55		0.24	0.10	0.10
	当被执行时	当导通时	2.8	2.1	1.1	0.24	0.10	0.10
		当不导通时	3.2	2.4	1.2	0.24	0.10	0.10
OR < >	当不被执行时	1.4	1.1	0.55		0.24	0.10	0.10
	当被执行时	当导通时	3.8	2.9	1.5	0.24	0.10	0.10
		当不导通时	2.8	2.1	1.1	0.24	0.10	0.10
LD >	当导通时	4.4	3.3	1.7		0.24	0.10	0.10
	当不导通时	3.6	2.7	1.4		0.24	0.10	0.10
AND >	当不被执行时	1.4	1.1	0.55		0.24	0.10	0.10
	当被执行时	当导通时	2.8	2.1	1.1	0.24	0.10	0.10
		当不导通时	3.2	2.4	1.2	0.24	0.10	0.10
OR >	当不被执行时	1.4	1.1	0.55		0.24	0.10	0.10
	当被执行时	当导通时	3.8	2.9	1.5	0.24	0.10	0.10
		当不导通时	2.8	2.1	1.1	0.24	0.10	0.10
LD < =	当导通时	4.4	3.3	1.7		0.24	0.10	0.10
	当不导通时	3.6	2.7	1.4		0.24	0.10	0.10
AND < =	当不被执行时	1.4	1.1	0.55		0.24	0.10	0.10
	当被执行时	当导通时	2.8	2.1	1.1	0.24	0.10	0.10
		当不导通时	3.2	2.4	1.2	0.24	0.10	0.10
OR < =	当不被执行时	1.4	1.1	0.55		0.24	0.10	0.10
	当被执行时	当导通时	3.8	2.9	1.5	0.24	0.10	0.10
		当不导通时	2.8	2.1	1.1	0.24	0.10	0.10
LD <	当导通时	4.4	3.3	1.7		0.24	0.10	0.10
	当不导通时	3.6	2.7	1.4		0.24	0.10	0.10
AND <	当不被执行时	1.4	1.1	0.55		0.24	0.10	0.10
	当被执行时	当导通时	2.8	2.1	1.1	0.24	0.10	0.10
		当不导通时	3.2	2.4	1.2	0.24	0.10	0.10
OR <	当不被执行时	1.4	1.1	0.55		0.24	0.10	0.10
	当被执行时	当导通时	3.8	2.9	1.5	0.24	0.10	0.10
		当不导通时	2.8	2.1	1.1	0.24	0.10	0.10
LD > =	当导通时	4.4	3.3	1.7		0.24	0.10	0.10
	当不导通时	3.6	2.7	1.4		0.24	0.10	0.10
AND > =	当不被执行时	1.4	1.1	0.55		0.24	0.10	0.10
	当被执行时	当导通时	2.8	2.1	1.1	0.24	0.10	0.10
		当不导通时	3.2	2.4	1.2	0.24	0.10	0.10
OR > =	当不被执行时	1.4	1.1	0.55		0.24	0.10	0.10
	当被执行时	当导通时	3.8	2.9	1.5	0.24	0.10	0.10
		当不导通时	2.8	2.1	1.1	0.24	0.10	0.10
LDD =	当导通时	5.0	3.8	1.9		0.55	0.24	0.24
	当不导通时	4.2	3.2	1.6		0.39	0.17	0.17
ANDD =	当不被执行时	1.4	1.1	0.55		0.39	0.17	0.17
	当被执行时	当导通时	3.4	2.6	1.3	0.55	0.24	0.24
		当不导通时	3.8	2.9	1.5	0.39	0.17	0.17

指令	条件(软元件)		处理时间(μs)						
			Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
ORD =	当不被执行时		1.4	1.1	0.55		0.39	0.17	0.17
	当被执行时	当导通时	4.4	3.3	1.7		0.55	0.24	0.24
		当不导通时	3.4	2.6	1.3		0.55	0.24	0.24
LDD < >	当导通时		5.0	3.8	1.9		0.55	0.24	0.24
	当不导通时		4.2	3.2	1.6		0.55	0.24	0.24
ANDD < >	当不被执行时		1.4	1.1	0.55		0.39	0.17	0.17
	当被执行时	当导通时	3.4	2.6	1.3		0.55	0.24	0.24
		当不导通时	3.8	2.9	1.5		0.55	0.24	0.24
ORD < >	当不被执行时		1.4	1.1	0.55		0.39	0.17	0.17
	当被执行时	当导通时	4.4	3.3	1.7		0.55	0.24	0.24
		当不导通时	3.4	2.6	1.3		0.55	0.24	0.24
LDD >	当导通时		3.8	2.9	1.5		0.55	0.24	0.24
	当不导通时		4.2	3.2	1.6		0.55	0.24	0.24
ANDD >	当不被执行时		1.4	1.1	0.55		0.39	0.17	0.17
	当被执行时	当导通时	2.8	2.1	1.1		0.55	0.24	0.24
		当不导通时	3.8	2.9	1.5		0.55	0.24	0.24
ORD >	当不被执行时		1.4	1.1	0.55		0.39	0.17	0.17
	当被执行时	当导通时	3.8	2.9	1.5		0.55	0.24	0.24
		当不导通时	3.4	2.6	1.3		0.55	0.24	0.24
LDD < =	当导通时		4.4	3.3	1.7		0.55	0.24	0.24
	当不导通时		3.6	2.7	1.4		0.55	0.24	0.24
ANDD < =	当不被执行时		1.4	1.1	0.55		0.39	0.17	0.17
	当被执行时	当导通时	3.4	2.6	1.3		0.55	0.24	0.24
		当不导通时	3.2	2.4	1.2		0.55	0.24	0.24
ORD < =	当不被执行时		1.4	1.1	0.55		0.39	0.17	0.17
	当被执行时	当导通时	4.4	3.3	1.7		0.55	0.24	0.24
		当不导通时	2.8	2.1	1.1		0.55	0.24	0.24
LDD <	当导通时		3.8	2.9	1.5		0.55	0.24	0.24
	当不导通时		4.2	3.2	1.6		0.55	0.24	0.24
ANDD <	当不被执行时		1.4	1.1	0.55		0.39	0.17	0.17
	当被执行时	当导通时	2.8	2.1	1.1		0.55	0.24	0.24
		当不导通时	3.8	2.9	1.5		0.55	0.24	0.24
ORD <	当不被执行时		1.4	1.1	0.55		0.39	0.17	0.17
	当被执行时	当导通时	3.8	2.9	1.5		0.55	0.24	0.24
		当不导通时	3.4	2.6	1.3		0.55	0.24	0.24
LDD > =	当导通时		4.4	3.3	1.7		0.55	0.24	0.24
	当不导通时		3.6	2.7	1.4		0.55	0.24	0.24
ANDD > =	当不被执行时		1.4	1.1	0.55		0.39	0.17	0.17
	当被执行时	当导通时	3.4	2.6	1.3		0.55	0.24	0.24
		当不导通时	3.2	2.4	1.2		0.55	0.24	0.24
ORD > =	当不被执行时		1.4	1.1	0.55		0.39	0.17	0.17
	当被执行时	当导通时	4.4	3.3	1.7		0.55	0.24	0.24
		当不导通时	2.8	2.1	1.1		0.55	0.24	0.24
LDE = *	单精度	当导通时	—	—	—	35	93	40	6.4
							14.9	6.4	
		当不导通时	—	—	—	87	92	40	6.4
							14.9	6.4	
	双精度	当导通时	235	177	89	—	93	40	—
							14.9	6.4	
		当不导通时	231	174	87	—	92	40	—
							14.9	6.4	

备注

※: Qn/QnH 的处理时间随 CPU 模块的系列号而变化。
顶部 : 系列号的首五位是“05031”或更小的数字
底部 : 系列号的首五位是“05032”或更大的数字
当指令不执行的条件得到满足时, 在顶部和底部之间没有差别。

指令	条件(软元件)		处理时间(μs)									
			Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH			
ANDE = *	单精度	当不被执行时		—	—	—	0.55	0.55	0.24			
		当被执行时	当导通时	—	—	—	34	93 14.9	40 6.4	6.4		
			当不导通时	—	—	—	86	92 14.9	40 6.4			
				当不被执行时		1.4	1.1	0.55	—		—	—
	双精度	当被执行时	当导通时	234	176	88	—	93 14.9	40 6.4	—		
			当不导通时	230	172	86	—	92 14.9	40 6.4		—	
				当不被执行时		1.4	1.1	0.55	—			—
		ORE = *	单精度	当不被执行时		—	—	—	0.55	0.55	0.24	
当被执行时	当导通时			—	—	—	35	93 14.9	40 6.4	6.4		
	当不导通时			—	—	—	86	92 14.9	40 6.4		6.4	
				当不被执行时		1.4	1.1	0.55	—			—
双精度	当被执行时		当导通时	234	176	88	—	93 14.9	40 6.4	—		
			当不导通时	230	172	86	—	92 14.9	40 6.4		—	
				当不被执行时		1.4	1.1	0.55	—			—
	LDE < > *		单精度	当导通时		—	—	—	35	92 14.9	40 6.4	6.4
当不导通时		—		—	—	88	92 14.9	40 6.4	6.4			
双精度		当导通时		231	174	87	—	92 14.9		40 6.4	—	
		当不导通时		234	176	88	—	92 14.9	40 6.4	—		
ANDE < > *		单精度	当不被执行时		—	—	—	0.55	0.55		0.24	
			当被执行时	当导通时	—	—	—	34	92 14.9	40 6.4	6.4	
				当不导通时	—	—	—	88	93 14.9	40 6.4		6.4
					当不被执行时		1.4	1.1	0.55	—		
	双精度	当被执行时	当导通时	230	172	86	—	92 14.9	40 6.4	—		
			当不导通时	234	176	88	—	92 14.9	40 6.4		—	
				当不被执行时		1.4	1.1	0.55	—			—
		ORE < > *	单精度	当不被执行时		—	—	—	0.55	0.55	0.24	
当被执行时	当导通时			—	—	—	35	93 14.9	40 6.4	6.4		
	当不导通时			—	—	—	88	92 14.9	40 6.4		6.4	
				当不被执行时		1.4	1.1	0.55	—			—
双精度	当被执行时		当导通时	231	174	87	—	93 14.9	40 6.4	—		
			当不导通时	234	176	88	—	92 14.9	40 6.4		—	
				当不被执行时		1.4	1.1	0.55	—			—

备注

*: Qn/QnH 的处理时间随 CPU 模块的系列号而变化。

顶部 : 系列号的首五位是“05031”或更小的数字

底部 : 系列号的首五位是“05032”或更大的数字

当不执行指令的条件得到满足时, 在顶部和底部之间没有差别。

指令	条件 (软元件)		处理时间(μs)						
			Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
LDE > *	单精度	当导通时	—	—	—	35	92	40	6.4
							14.9	6.4	
	单精度	当不导通时	—	—	—	88	92	40	6.4
							14.9	6.4	
	双精度	当导通时	231	174	87	—	92	40	—
							14.9	6.4	
LDE > *	单精度	当导通时	—	—	—	35	92	40	6.4
							14.9	6.4	
	单精度	当不导通时	—	—	—	88	92	40	6.4
							14.9	6.4	
	双精度	当导通时	231	174	87	—	92	40	—
							14.9	6.4	
AND > *	单精度	当不被执行时	—	—	—	0.55	0.55	0.24	0.24
							0.55	0.24	
	单精度	当被执行时	—	—	—	34	92	40	6.4
							14.9	6.4	
	单精度	当不导通时	—	—	—	88	93	40	6.4
							14.9	6.4	
AND > *	双精度	当不被执行时	1.4	1.1	0.55	—	—	—	—
							—	—	
	双精度	当被执行时	230	172	86	—	92	40	—
							14.9	6.4	
	双精度	当不导通时	234	176	88	—	92	40	—
							14.9	6.4	
OR > *	单精度	当不被执行时	—	—	—	0.55	0.55	0.24	0.24
							0.55	0.24	
	单精度	当被执行时	—	—	—	34	93	40	6.4
							14.9	6.4	
	单精度	当不导通时	—	—	—	88	92	40	6.4
							14.9	6.4	
LDE < = *	单精度	当导通时	—	—	—	34	93	40	6.4
							14.9	6.4	
	单精度	当不导通时	—	—	—	88	92	40	6.4
							14.9	6.4	
	双精度	当导通时	235	177	89	—	93	40	—
							14.9	6.4	
LDE < = *	双精度	当不导通时	231	174	87	—	92	40	—
							14.9	6.4	

备注

*: Qn/QnH 的处理时间随 CPU 模块的系列号而变化。

顶部 : 系列号的首五位是“05031”或更小的数字

底部 : 系列号的首五位是“05032”或更大的数字

当指令不执行的条件得到满足时, 在顶部和底部之间没有差别。

指令	条件(软元件)		处理时间(μs)								
			Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH		
ANDE < = *	单精度	当不被执行时		—	—	—	0.55	0.55	0.24	0.24	
		当被执行时	当导通时	—	—	—	34	92 14.9	40 6.4	6.4	
			当不导通时	—	—	—	86	92 14.9	40 6.4		6.4
		双精度	当不被执行时		1.4	1.1	0.55	—	—	—	
	当被执行时		当导通时	234	176	88	—	92 14.9	40 6.4	—	
			当不导通时	230	172	86	—	92 14.9	40 6.4		—
	ORE < = *		单精度	当不被执行时		—	—	—	0.55	0.55	
		当被执行时		当导通时	—	—	—	34	92 14.9	40 6.4	6.4
				当不导通时	—	—	—	86	92 14.9	40 6.4	
双精度		当不被执行时		1.4	1.1	0.55	—	—	—	—	
		当被执行时	当导通时	234	176	88	—	92 14.9	40 6.4	—	
			当不导通时	230	172	86	—	92 14.9	40 6.4		—
		LDE < *	单精度	当导通时		—	—	—	35	92 14.9	
当不导通时				—	—	—	88	92 14.9	40 6.4	6.4	
双精度	当导通时		231	174	87	—	92 14.9	40 6.4	—		
	当不导通时		234	176	88	—	92 14.9	40 6.4		—	
ANDE < *	单精度		当不被执行时		—	—	—	0.55	0.55		0.24
			当被执行时	当导通时	—	—	—	34	92 14.9	40 6.4	6.4
		当不导通时		—	—	—	88	92 14.9	40 6.4	6.4	
		双精度	当不被执行时		1.4	1.1	0.55	—	—		—
	当被执行时		当导通时	230	172	86	—	92 14.9	40 6.4	—	
			当不导通时	234	176	88	—	92 14.9	40 6.4		—

备注

※: Qn/QnH 的处理时间随 CPU 模块的系列号而变化。
顶部 : 系列号的首五位是“05031”或更小的数字
底部 : 系列号的首五位是“05032”或更大的数字
当指令不执行的条件得到满足时, 在顶部和底部之间没有差别。

指令	条件(软元件)		处理时间(μs)								
			Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH		
ORE < *	单精度	当不被执行时		—	—	—	0.55	0.55	0.24	0.24	
		当被执行时	当导通时	—	—	—	34	93	40	6.4	
			当不导通时	—	—	—	88	92	40		
				—	—	—	—	14.9	6.4	6.4	
		双精度	当不被执行时		1.4	1.1	0.55	—	—	—	—
	当被执行时		当导通时	231	174	87	—	93	40	—	
			当不导通时	234	176	88	—	14.9	6.4		
				—	—	—	—	92	40	—	
	LDE > = *		单精度	当导通时		—	—	—	35	93	40
		当不导通时		—	—	—	87	92	40	6.4	
双精度		当导通时		235	177	89	—	14.9	6.4		—
		当不导通时		231	174	87	—	93	40	—	
		—	—	—	—	14.9	6.4	—			
		ANDE > = *	单精度	当不被执行时		—	—	—	0.55	0.55	0.24
当被执行时				当导通时	—	—	—	34	92	40	6.4
				当不导通时	—	—	—	87	14.9	6.4	
					—	—	—	—	92	40	6.4
双精度				当不被执行时		1.4	1.1	0.55	—	—	—
	当被执行时		当导通时	234	176	88	—	92	40	—	
			当不导通时	231	174	87	—	14.9	6.4		
				—	—	—	—	92	40	—	
	ORE > = *		单精度	当不被执行时		—	—	—	0.55	0.55	0.24
当被执行时				当导通时	—	—	—	34	92	40	6.4
		当不导通时		—	—	—	86	14.9	6.4		
				—	—	—	—	92	40	6.4	
双精度		当不被执行时		1.4	1.1	0.55	—	—	—	—	
		当被执行时	当导通时	234	176	88	—	92	40	—	
			当不导通时	230	172	86	—	14.9	6.4		
				—	—	—	—	92	40	—	





































备注

*: Qn/QnH 的处理时间随 CPU 模块的系列号而变化。

顶部 : 系列号的首五位是“05031”或更小的数字

底部 : 系列号的首五位是“05032”或更大的数字

当指令不执行的条件得到满足时, 在顶部和底部之间没有差别。

指令	条件(软元件)	处理时间(μs)						
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
LD \$ =	当导通时	97	73	37		38	16	16
	当不导通时	81	61	31		34	15	15
AND \$ =	当不被执行时	1.4	1.1	0.55		0.56	0.23	0.23
	当被执行时	当导通时	96	72	36	39	17	17
		当不导通时	81	61	31	32	14	14
	当不被执行时	1.4	1.1	0.55		0.56	0.24	0.24
OR \$ =	当被执行时	当导通时	97	73	37	40	17	17
		当不导通时	80	60	30	33	14	14
	当不被执行时	1.4	1.1	0.55		0.56	0.24	0.24
LD \$ < >	当导通时	83	62	31		32	14	14
	当不导通时	97	73	37		40	17	17
AND \$ < >	当不被执行时	1.4	1.1	0.55		0.56	0.23	0.23
	当被执行时	当导通时	80	60	30	33	14	14
		当不导通时	96	72	36	39	17	17
	当不被执行时	1.4	1.1	0.55		0.56	0.24	0.24
OR \$ < >	当被执行时	当导通时	81	61	31	32	14	14
		当不导通时	96	72	36	39	17	17
	当不被执行时	1.4	1.1	0.55		0.56	0.24	0.24
LD \$ >	当导通时	83	62	31		32	14	14
	当不导通时	97	73	37		40	17	17
AND \$ >	当不被执行时	1.4	1.1	0.55		0.56	0.23	0.23
	当被执行时	当导通时	80	60	30	33	14	14
		当不导通时	96	72	36	39	17	17
	当不被执行时	1.4	1.1	0.55		0.56	0.24	0.24
OR \$ >	当被执行时	当导通时	81	61	31	32	14	14
		当不导通时	96	72	36	39	17	17
	当不被执行时	1.4	1.1	0.55		0.56	0.24	0.24
LD \$ < =	当导通时	97	73	37		40	17	17
	当不导通时	81	61	31		32	14	14
AND \$ < =	当不被执行时	1.4	1.1	0.55		0.56	0.23	0.23
	当被执行时	当导通时	96	72	36	39	17	17
		当不导通时	81	61	31	32	14	14
	当不被执行时	1.4	1.1	0.55		0.56	0.24	0.24
OR \$ < =	当被执行时	当导通时	97	73	37	40	17	17
		当不导通时	80	60	30	33	14	14
	当不被执行时	1.4	1.1	0.55		0.56	0.24	0.24
LD \$ <	当导通时	81	61	31		32	14	14
	当不导通时	97	73	37		40	17	17
AND \$ <	当不被执行时	1.4	1.1	0.55		0.56	0.23	0.23
	当被执行时	当导通时	80	60	30	32	14	14
		当不导通时	96	72	36	39	16	16
	当不被执行时	1.4	1.1	0.55		0.56	0.24	0.24
OR \$ <	当被执行时	当导通时	81	61	31	32	14	14
		当不导通时	96	72	36	39	16	16
	当不被执行时	1.4	1.1	0.55		0.56	0.24	0.24
LD \$ > =	当导通时	97	73	37		40	17	17
	当不导通时	81	61	31		32	14	14
AND \$ > =	当不被执行时	1.4	1.1	0.55		0.56	0.23	0.23
	当被执行时	当导通时	96	72	36	39	16	16
		当不导通时	81	61	31	32	14	14
	当不被执行时	1.4	1.1	0.55		0.56	0.24	0.24
OR \$ > =	当被执行时	当导通时	97	73	37	39	17	17
		当不导通时	80	60	30	32	14	14
	当不被执行时	1.4	1.1	0.55		0.56	0.24	0.24
BKCM P =    n	n = 1	120	90	45		48	21	21
BKCM P = P    n	n = 96	367	276	138		142	61	61
BKCM P < >    n	n = 1	123	92	46		48	21	21
BKCM P < > P    n	n = 96	346	260	130		150	65	65
BKCM P >    n	n = 1	123	92	96		48	21	21
BKCM P > P    n	n = 96	366	275	138		142	61	61
BKCM P > =    n	n = 1	121	91	46		48	21	21
BKCM P > = P    n	n = 96	386	290	145		150	65	65
BKCM P <    n	n = 1	121	91	96		48	21	21
BKCM P < P    n	n = 96	366	275	138		158	68	68
BKCM P < =    n	n = 1	121	91	46		48	21	21
BKCM P < = P    n	n = 96	348	261	131		150	65	65

指令	条件(软元件)	处理时间(μs)						
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
+ (S) (D) + P (S) (D)	当被执行时	2.4	1.8	0.9		0.39	0.17	0.17
+ (S1) (S2) (D) + P (S1) (S2) (D)	当被执行时	2.7	2.0	1.0		0.47	0.20	0.20
- (S) (D) - P (S) (D)	当被执行时	2.4	1.8	0.9		0.39	0.17	0.17
- (S1) (S2) (D) - P (S1) (S2) (D)	当被执行时	2.6	2.0	1.0		0.47	0.20	0.20
D + (S) (D) D + P (S) (D)	当被执行时	2.8	2.1	1.1		0.71	0.31	0.31
D + (S1) (S2) (D) D + P (S1) (S2) (D)	当被执行时	3.2	2.4	1.2		0.79	0.34	0.34
D - (S) (D) D - P (S) (D)	当被执行时	2.8	2.1	1.1		0.71	0.30	0.30
D - (S1) (S2) (D) D - P (S1) (S2) (D)	当被执行时	3.2	2.4	1.2		0.79	0.34	0.34
* (S1) (S2) (D) * P (S1) (S2) (D)	当被执行时	2.8	2.1	1.1		0.47	0.20	0.20
/ (S1) (S2) (D) /P (S1) (S2) (D)		6.8	5.1	2.6		2.7	1.2	1.2
D * (S1) (S2) (D) D * P (S1) (S2) (D)		20	15	7.5		7.9	3.4	3.4
D / (S1) (S2) (D) D / P (S1) (S2) (D)		36	27	13.5		14	6.1	6.1
B + (S) (D) B + P (S) (D)		5.5	4.1	2.1		2.2	1.0	1.0
B + (S1) (S2) (D) B + P (S1) (S2) (D)		13	9.6	4.8		5.0	2.2	2.2
B - (S) (D) B - P (S) (D)		5.2	3.9	2.0		2.0	0.9	0.9
B - (S1) (S2) (D) B - P (S1) (S2) (D)		13	9.4	4.7		4.9	2.1	2.1
DB + (S) (D) DB + P (S) (D)		29	22	11		12	5.0	5.0
DB + (S1) (S2) (D) DB + P (S1) (S2) (D)		32	24	12		12	5.3	5.3
DB - (S) (D) DB - P (S) (D)		29	22	11		11	4.8	4.8
DB - (S1) (S2) (D) DB - P (S1) (S2) (D)		32	24	12		12	5.2	5.2
B * (S1) (S2) (D) B * P (S1) (S2) (D)		9.4	7.1	3.6		3.7	1.6	1.6
B / (S1) (S2) (D) B / P (S1) (S2) (D)		9.4	7.1	3.6		3.8	1.6	1.6
DB * (S1) (S2) (D) DB * P (S1) (S2) (D)		62	46	23		24	10	10
DB / (S1) (S2) (D) DB / P (S1) (S2) (D)		69	52	26		27	12	12

指令	条件(软元件)		处理时间(μs)						
			Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
E + \textcircled{S} \textcircled{D} E + P \textcircled{S} \textcircled{D}	单精度	$\textcircled{S} = 0, \textcircled{D} = 0$	—	—	—	35	1.8	0.78	0.78
		$\textcircled{S} = 2^{127}, \textcircled{D} = 2^{127}$	—	—	—	35	1.8	0.78	0.78
	双精度	$\textcircled{S} = 0, \textcircled{D} = 0$	54	40	20	—	203	87	—
		$\textcircled{S} = 2^{127}, \textcircled{D} = 2^{127}$	524	394	197	—	203	87	—
E + $\textcircled{S_1} \textcircled{S_2} \textcircled{D}$ E + P $\textcircled{S_1} \textcircled{S_2} \textcircled{D}$	单精度	$\textcircled{S} = 0, \textcircled{D} = 0$	—	—	—	35	2.4	1.1	1.1
		$\textcircled{S} = 2^{127}, \textcircled{D} = 2^{127}$	—	—	—	35	2.4	1.1	1.1
	双精度	$\textcircled{S} = 0, \textcircled{D} = 0$	54	40	20	—	209	90	—
		$\textcircled{S} = 2^{127}, \textcircled{D} = 2^{127}$	524	394	197	—	209	90	—
E - \textcircled{S} \textcircled{D} E - P \textcircled{S} \textcircled{D}	单精度	$\textcircled{S} = 0, \textcircled{D} = 0$	—	—	—	35	1.8	0.78	0.78
		$\textcircled{S} = 2^{127}, \textcircled{D} = 2^{127}$	—	—	—	35	1.8	0.78	0.78
	双精度	$\textcircled{S} = 0, \textcircled{D} = 0$	54	40	20	—	202	87	—
		$\textcircled{S} = 2^{127}, \textcircled{D} = 2^{127}$	515	387	194	—	202	87	—
E - $\textcircled{S_1} \textcircled{S_2} \textcircled{D}$ E - P $\textcircled{S_1} \textcircled{S_2} \textcircled{D}$	单精度	$\textcircled{S_1} = 0, \textcircled{S_2} = 0$	—	—	—	35	2.4	1.1	1.1
		$\textcircled{S_1} = 2^{127}, \textcircled{S_2} = 2^{127}$	—	—	—	36	2.4	1.1	1.1
	双精度	$\textcircled{S_1} = 0, \textcircled{S_2} = 0$	55	41	21	—	210	90	—
		$\textcircled{S_1} = 2^{127}, \textcircled{S_2} = 2^{127}$	520	391	146	—	210	90	—
E * $\textcircled{S_1} \textcircled{S_2} \textcircled{D}$ E * P $\textcircled{S_1} \textcircled{S_2} \textcircled{D}$	单精度	$\textcircled{S_1} = 0, \textcircled{S_2} = 0$	—	—	—	35	2.4	1.1	1.1
		$\textcircled{S_1} = 2^{126}, \textcircled{S_2} = 2^{127}$	—	—	—	36	2.4	1.1	1.1
	双精度	$\textcircled{S_1} = 0, \textcircled{S_2} = 0$	55	41	21	—	222	96	—
		$\textcircled{S_1} = 2^{126}, \textcircled{S_2} = 2^{127}$	567	426	218	—	222	96	—
E / $\textcircled{S_1} \textcircled{S_2} \textcircled{D}$ E / P $\textcircled{S_1} \textcircled{S_2} \textcircled{D}$	单精度	$\textcircled{S_1} = 0, \textcircled{S_2} = 1$	—	—	—	37	12	5.2	5.2
		$\textcircled{S_1} = 2^{127}, \textcircled{S_2} = -2^{126}$	—	—	—	38	12	5.2	5.2
	双精度	$\textcircled{S_1} = 0, \textcircled{S_2} = 1$	149	112	56	—	369	159	—
		$\textcircled{S_1} = 2^{127}, \textcircled{S_2} = -2^{126}$	1109	834	417	—	369	159	—

指令	条件(软元件)		处理时间(μs)					
			Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH QnPH
\$ + \textcircled{S} \textcircled{D} \$ + P \textcircled{S} \textcircled{D}			179	134	67		68	29 29
\$ + $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D} \$ + P $\textcircled{S1}$ $\textcircled{S2}$ \textcircled{D}			206	155	78		81	35 35
INC INCP			1.9	1.4	0.7		0.32	0.14 0.14
DINC DINCP			2.3	1.7	0.9		0.47	0.20 0.20
DEC DECP			1.9	1.4	0.7		0.32	0.14 0.14
DDEC DDECP			2.3	1.7	0.9		0.47	0.20 0.20
BCD BCDP			2.7	2.0	1.0		1.1	0.48 0.48
DBCD DBCDP			7.9	5.9	3.0		3.2	1.4 1.4
BIN BINP			2.7	2.0	1.0		1.0	0.44 0.44
DBIN DBINP			4.8	3.6	1.8		1.9	0.82 0.82
INT INTP	单精度	$\textcircled{S} = 0$	—	—	—		3.2	1.4 1.4
		$\textcircled{S} = 32766.5$	—	—	—		3.2	1.4 1.4
	双精度	$\textcircled{S} = 0$	20	15	7.5		22	9.3 —
		$\textcircled{S} = 32766.5$	54	40	20		22	9.3 —
DINT DINTP	单精度	$\textcircled{S} = 0$	—	—	—		2.5	1.1 1.1
		$\textcircled{S} = 1234567890.3$	—	—	—		2.5	1.1 1.1
	双精度	$\textcircled{S} = 0$	20	15	7.5		24	10 —
		$\textcircled{S} = 1234567890.3$	59	44	22		24	10 —
FLT FLTP	单精度	$\textcircled{S} = 0$	—	—	—		2.1	0.92 0.92
		$\textcircled{S} = 7FFF_{\text{H}}$	—	—	—		2.1	0.92 0.92
	双精度	$\textcircled{S} = 0$	27	20	10		22	9.6 —
		$\textcircled{S} = 7FFF_{\text{H}}$	55	41	21		22	9.6 —
DFLT DFLTP	单精度	$\textcircled{S} = 0$	—	—	—		2.1	0.88 0.88
		$\textcircled{S} = 7FFFFFFF_{\text{H}}$	—	—	—		2.1	0.88 0.88
	双精度	$\textcircled{S} = 0$	28	21	11		26	11 —
		$\textcircled{S} = 7FFFFFFF_{\text{H}}$	56	42	21		26	11 —

指令	条件(软元件)	处理时间(μs)					
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH QnPH
DBL DBLP		12	8.6	4.3		4.5	1.9 1.9
WORD WORDP		12	9.0	4.5		4.7	2.0 2.0
GRY GRYP		12	9.0	4.5		4.7	2.0 2.0
DGRY DGRYP		14	10	5.0		5.3	2.3 2.3
GBIN GBINP		46	34	17		18	7.7 7.7
DGBIN DGBINP		83	62	31		32	14 14
NEG NEGP		9.3	7	3.5		3.6	1.6 1.6
DNeg DNEGP		11	8.2	4.1		4.3	1.8 1.8
ENeg ENE GP		9.8	7.4	3.7		3.9	1.7 1.7
BKBCD (S) (D) n	n = 1	102	76	38		38	17 17
BKBCDP (S) (D) n	n = 96	272	204	102		99	43 43
BKBIN (S) (D) n	n = 1	102	76	38		38	17 17
BKBINP (S) (D) n	n = 96	272	204	102		99	43 43
MOV MOVP	(S) = D0, (D) = D1	0.7	0.5	0.3		0.24	0.10 0.10
	(S) = D0, (D) = J1~W1	392 *	305 *	176 *		—	— —
		391 *	299 *	165 *		—	— —
		—	—	—		140 *	60 * 60 *
DMOV DMOVP	(S) = D0, (D) = D1	2.4	1.8	0.9		0.47	0.20 0.20
	(S) = D0, (D) = J1~W1	400 *	313 *	183 *		—	— —
		395 *	301 *	167 *		—	— —
		—	—	—		147 *	64 * 64 *
EMOV EMOVP		12	8.6	4.3		0.63	0.27 0.27
\$ MOV \$ MOVP		100	75	38		40	17 17
CML CMLP		2.0	1.5	0.8		0.40	0.17 0.17
DCML DCMLP		2.4	1.8	0.9		0.55	0.24 0.24

备注

- 1) *: 上面的一行表示使用 A38B/A1S38B 和扩展基板时的处理时间。
中间一行表示使用 A38HB/A1S38HB 时的处理时间。
下面一行表示使用 Q312B 时的处理时间。























































指令	条件(软元件)	处理时间(μs)						
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
BMOV \textcircled{S} \textcircled{D} n	n = 1	43	32	16		17	7.1	7.1
BMOVP \textcircled{S} \textcircled{D} n	n = 96	81	61	31		32	14	14
FMOV \textcircled{S} \textcircled{D} n	n = 1	18	13	6.5		6.7	2.9	2.9
FMOVP \textcircled{S} \textcircled{D} n	n = 96	36	27	14		14	6.1	6.1
XCH XCHP DXCH DXCHP		3.1	2.3	1.2		1.3	0.54	0.54
BXCH \textcircled{D} \textcircled{D} n	n = 1	77	58	29		31	13	13
BXCHP \textcircled{D} \textcircled{D} n	n = 96	213	160	80		84	36	36
SWAP SWAPP		9.2	6.9	3.5		3.7	1.6	1.6
CJ		7.8	5.8	2.9		3.2	1.4	1.4
SCJ		7.8	5.8	2.9		3.2	1.4	1.4
JMP		8.0	6.0	3.0		3.2	1.4	1.4
GOEND		2.0	1.5	0.75		0.39	0.34	0.34
DI		2.3	1.7	0.9		0.95	0.41	0.41
EI		3.1	2.3	1.2		1.3	0.54	0.54
IMASK		8.1	6.5	3.3		11	4.6	4.6
IRET		4.0	3.0	1.5		1.6	0.68	0.68
RFS	n = 1	31.3	23.4	11.7		6.7	4.7	4.7
RFSP	n = 96	97.6	72.8	36.4		19	13	13
UDCNT1		42.6	31.8	15.9		15	6.5	6.5
UDCNT2		44.6	33.3	16.7		16	6.8	6.8
TTMR		25.9	19.3	9.7		10	4.4	4.4
STMR		41.7	31.1	15.6		20	7.1	7.1
ROTC		66.1	49.3	24.7		26	11	11
RAMP		45.4	33.9	17.0		18	7.7	7.7
SPD		48.9	36.5	18.3		19	8.3	8.3
PLSY		26.9	20.1	10.1		10	4.5	4.5
PWM		32.8	24.5	12.3		9.1	3.9	3.9
MTR		29.2	21.8	10.9		11	4.9	4.9

(3) 应用指令

指令不执行时的处理时间按照如下方式计算:

Q2ACPU (S1), Q2ASCPU (S1) $0.20 \times (\text{每条指令的执行步数} + 1) \mu\text{s}$
 Q3ACPU $0.15 \times (\text{每条指令的执行步数} + 1) \mu\text{s}$
 Q2ASHCPU (S1), Q4ACPU, Q4ARCPU $0.075 \times (\text{每条指令的执行步数} + 1) \mu\text{s}$
 Q02CPU $0.079 \times (\text{每条指令的执行步数} + 1) \mu\text{s}$
 Q02HCPU, Q06HCPU, Q12HCPU, Q25HCPU
 Q12PHCPU, Q25PHCPU $0.034 \times (\text{每条指令的执行步数} + 1) \mu\text{s}$

指令	条件(软元件)	处理时间(μs)						
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
WAND $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix}$	当被执行时	2.4	1.8	0.9		0.39	0.17	0.17
WANDP $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix}$	当被执行时	9.5	7.1	3.6		0.47	0.20	0.20
DAND $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix}$	当被执行时	3.0	2.3	1.2		0.71	0.31	0.31
DANDP $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix}$	当被执行时	19	14	7.0		0.79	0.34	0.34
BKAND $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix} n$	n = 1	89	67	34		36	16	16
BKANDP $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix} n$	n = 96	184	138	69		74	32	32
WOR $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix}$	当被执行时	2.4	1.8	0.9		0.40	0.17	0.17
WORP $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix}$	当被执行时	9.5	7.1	3.6		0.47	0.20	0.20
DOR $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix}$	当被执行时	3.0	2.3	1.2		0.71	0.31	0.31
DORP $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix}$	当被执行时	19	14	7.0		0.79	0.34	0.34
BKOR $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix} n$	n = 1	89	67	34		36	16	16
BKORP $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix} n$	n = 96	184	138	69		74	32	32
WXOR $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix}$	当被执行时	2.4	1.8	0.9		0.39	0.17	0.17
WXORP $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix}$	当被执行时	17.2	7.1	3.6		0.47	0.20	0.20
DXOR $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix}$	当被执行时	3.0	2.3	1.2		0.71	0.31	0.31
DXORP $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix}$	当被执行时	19	14	7.0		0.79	0.34	0.34
BKXOR $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix} n$	n = 1	89	67	34		36	16	16
BKXORP $\begin{smallmatrix} \text{S} & \text{D} \\ \text{S} & \text{D} \end{smallmatrix} n$	n = 96	184	138	69		74	32	32

指令	条件(软元件)	处理时间(μs)						
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
WXNR   WXNRP  	当被执行时	2.4	1.8	0.9		0.40	0.17	0.17
WXNR    WXNRP   	当被执行时	9.5	7.1	3.6		0.47	0.20	0.20
DNXR   DNXRP  	当被执行时	3.0	2.3	1.2		0.71	0.31	0.31
DNXR    DNXRP   	当被执行时	24	18	9		0.79	0.34	0.34
BKNXOR    n	n = 1	89	67	34		36	16	16
BKNXORP    n	n = 96	184	138	69		74	32	32
ROR  n	n = 1	5.0	3.8	1.9		2.0	0.85	0.85
RORP  n	n = 15	5.0	3.8	1.9		2.0	0.85	0.85
RCR  n	n = 1	4.0	3.0	1.5		1.6	0.68	0.68
RCRP  n	n = 15	4.0	3.0	1.5		1.6	0.68	0.68
ROL  n	n = 1	5.0	3.8	1.9		2.0	0.85	0.85
ROLP  n	n = 15	5.0	3.8	1.9		2.0	0.85	0.85
RCL  n	n = 1	4.0	3.0	1.5		1.6	0.68	0.68
RCLP  n	n = 15	4.0	3.0	1.5		1.6	0.68	0.68
DROR  n	n = 1	9.8	7.4	3.7		3.9	1.7	1.7
DRORP  n	n = 31	10	7.8	3.9		4.0	1.7	1.7
DRCR  n	n = 1	11	8.1	4.1		4.3	1.8	1.8
DRCRP  n	n = 31	11	8.3	4.2		4.3	1.9	1.9
DROL  n	n = 1	9.8	7.4	3.7		3.9	1.7	1.7
DROLP  n	n = 31	10	7.8	3.9		4.0	1.7	1.7
DRCL  n	n = 1	11	8.1	4.1		4.3	1.8	1.8
DRCLP  n	n = 31	11	8.3	4.2		4.3	1.9	1.9
SFR  n	n = 1	4.4	3.3	1.7		1.7	0.75	0.75
SFRP  n	n = 15	5.0	3.8	1.9		2.0	0.85	0.85
SFL  n	n = 1	4.4	3.3	1.7		1.7	0.75	0.75
SFLP  n	n = 15	5.0	3.8	1.9		2.0	0.85	0.85
BSFLR  n	n = 1	51	38	19		20	8.6	8.6
BSFLRP  n	n = 96	60	45	23		24	10	10
BSFL  n	n = 1	49	37	19		20	8.5	8.5
BSFLP  n	n = 96	58	44	22		23	10	10
DSFR  n	n = 1	3.6	2.6	1.3		1.3	0.58	0.58
DSFRP  n	n = 96	63	47	24		25	11	11
DSFL  n	n = 1	3.6	2.6	1.3		1.3	0.58	0.58
DSFLP  n	n = 96	65	49	25		26	11	11

指令	条件(软元件)	处理时间(μs)					
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH QnPH
BSET (D) n	n = 1	20	15	7.5		7.6	3.3
BSTP (D) n	n = 15	20	15	7.5		7.6	3.3
BRST (D) n	n = 1	20	15	7.5		7.6	3.3
BRSTP (D) n	n = 15	20	15	7.5		7.6	3.3
TEST (S) (D) (S) (D)		21	16	8.0		8.2	3.5
TESTP (S) (D) (S) (D)							
DTEST (S) (D) (S) (D)		24	18	9.0		9.2	3.9
DTESTP (S) (D) (S) (D)							
BKRST (S) n	n = 1	45	34	17		18	7.8
BKRSTP (S) n	n = 96	49	37	19		19	8.2
SER (S) (D) (S) (D) n	n = 1	58	44	22		22	9.6
SERP (S) (D) (S) (D) n	n = 96	57	43	21		21	8.9
		293	220	110		115	49
		340	256	128		133	57
DSER (S) (D) (S) (D) n	n = 1	61	46	23		23	9.9
DSERP (S) (D) (S) (D) n	n = 96	58	44	22		23	9.7
		354	266	133		142	61
		354	266	133		132	57
SUM	(S) = 0	9.8	7.4	3.7		3.9	1.7
SUMP	(S) = FFFF						
DSUM	(S) = 0	12	9.0	4.5		4.7	2.0
DSUMP	(S) = FFFFFFFFh	31	23	12		12	5.0
DECO (S) (D) n	n = 2	48	36	18		20	8.6
DECOP (S) (D) n	n = 8	62	47	24		27	12
ENCO (S) (D) n	n = 2	52	39	20		21	9.1
ENCOP (S) (D) n	n = 8	52	39	20		21	9.1
		65	49	25		28	12
		65	49	25		26	11
SEG		3.2	2.4	1.2		1.3	0.54
SEGP							
DIS (S) (D) n	n = 1	46	34	17		18	7.7
DISP (S) (D) n	n = 4	51	38	19		19	8.3
UNI (S) (D) n	n = 1	53	40	20		21	8.9
UNIP (S) (D) n	n = 4	57	43	22		23	9.7
NDIS (S) (D) (S) (D)		104	78	39		41	18
NDISP (S) (D) (S) (D)							
NUNI (S) (D) (S) (D)		105	79	40		42	18
NUNIP (S) (D) (S) (D)							
WTOB (S) (D) n	n = 1	125	94	47		47	20
WTOBP (S) (D) n	n = 96	257	193	97		99	43
BTOW (S) (D) n	n = 1	121	91	46		45	19
BTOWP (S) (D) n	n = 96	233	175	88		89	38

指令	条件(软元件)	处理时间(μs)					
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH QnPH
MAX (S) (D) n	n = 1	43	32	16		17	7.1
MAXP (S) (D) n	n = 96	318	239	120		136	59
MIN (S) (D) n	n = 1	43	32	16		17	7.1
MINP (S) (D) n	n = 96	436	326	163		159	69
DMAX (S) (D) n	n = 1	71	53	27		27	12
DMAXP (S) (D) n	n = 96	427	321	161		181	78
DMIN (S) (D) n	n = 1	71	53	27		27	12
DMINP (S) (D) n	n = 96	268	201	101		112	48
SORT (S) n (S) (S) (S)	n = 1	43	32	16		16	7.1
	n = 96	40 *	30 *	15 *		14	6.2
DSORT (S) n (S) (S) (S) (S)	n = 1	44	33	17		17	7.1
	n = 96	43 *	32 *	16 *		16	6.8
WSUM (S) (D) n	n = 1	41.5	31.1	15.6		16.4	7.1
WSUMP (S) (D) n	n = 96	173.2	129.9	65		68.4	29.5
DWSUM (S) (D) n	n = 1	47.9	35.9	18		18.9	8.2
DWSUMP (S) (D) n	n = 96	330	247.5	123.8		130.4	56.1
FOR n	n = 0	5.2	3.9	2.0		2.3	1.0
NEXT		8.0	6.0	3.0		3.3	1.4
BREAK							
BREAKP		26	19	9.5		11	4.6
CALL Pn	内部文件指针	5.1	3.8	1.9		2.1	0.88
CALLP Pn	公共指针	85	64	32		33	14
CALL Pn (S) 到 (S)		348	261	131		135	58
CALLP Pn (S) 到 (S)							
RET	返回到初始化程序	7.5	5.6	2.8		2.9	1.3
	返回到其它程序	51	38	19		20	8.5
FCALL Pn	内部文件指针	8.8	6.6	3.3		3.6	1.6
FCALLP Pn	公共指针	48	36	18		20	8.7
FCALL Pn (S) 到 (S)		338	254	127		134	57
FCALLP Pn (S) 到 (S)							
ECALL * Pn							
ECALLP * Pn		187	140	70		77	33
* :程序名称							
ECALL * Pn (S) 到 (S)		515	387	144		162	70
ECALLP * Pn (S) 到 (S)							
* :程序名称							
EFCALL * Pn							
EFCALLP * Pn		188	141	71		78	34
* :程序名称							
EFCALL * Pn (S) 到 (S)		516	388	194		200	86
EFCALLP * Pn (S) 到 (S)							
* :程序名称							

* :表示为执行完指令所需要的扫描时间的延长

指令	条件(软元件)		处理时间(μs)					
			Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH QnPH
COM			137	103	52		55	16 16
IX			31	23	12		12	5.2 5.2
IXEND			12	8.9	4.5		4.7	2.0 20
IXDEV +	触点数为 1		127	95	46		48	21 21
IXSET	触点数为 14		238	179	85		93	40 40
FIFW	数据点数为 0		27	20	10		11	4.5 4.5
FIFWP	数据点数为 96		27	20	10		11	4.5 4.5
FIFR	数据点数为 1		34	25	13		13	5.6 5.6
FIFRP	数据点数为 96		79	59	30		32	14 14
FPOP	数据点数为 1		46	34	17		16	7.0 7.0
FPOPP	数据点数为 96		46	34	17		16	7.0 7.0
FINS	数据点数为 0		48	36	18		20	8.4 8.4
FINSP	数据点数为 96		96	72	36		36	15 15
FDEL	数据点数为 1		47	35	18		19	7.5 7.5
FDELP	数据点数为 96		97	73	37		39	15 15
FROM n1 n2 ⑤ n3 FROMP n1 n2 ⑤ n3 *	n3 = 1		253	217	160		—	— —
			252	210	154		—	— —
			—	—	—		47	22 22
	n3 = 1000		4514	4286	4150		—	— —
			2855	2127	2038		—	— —
			—	—	—		476	437 437
DFRO n1 n2 ⑤ n3 DFROP n1 n2 ⑤ n3 *	n3 = 1		260	221	165		—	— —
			257	214	156		—	— —
			—	—	—		51	24 24
	n3 = 500		4543	4271	4082		—	— —
			2883	2129	2064		—	— —
			—	—	—		478	437 437
TO n1 n2 ⑤ n3 TOP n1 n2 ⑤ n3 *	n3 = 1		276	217	162		—	— —
			254	211	154		—	— —
			—	—	—		48	20 20
	n3 = 1000		4500	4319	4188		—	— —
			2878	2155	2043		—	— —
			—	—	—		479	412 412
DTO n1 n2 ⑤ n3 DTOP n1 n2 ⑤ n3 *	n3 = 1		260	221	165		—	— —
			257	216	157		—	— —
			—	—	—		50	23 23
	n3 = 500		4471	4315	4198		—	— —
			2819	2172	2062		—	— —
			—	—	—		457	416 416
PR	SM7010N	变量 1 个符	83	62	31		33	11 11
		变量 32 个字符	123	92	46		48	18 18
	SM7010FF		54	40	20		21	7.8 7.8
PRC			400	301	151		181	16 16
LED	当被显示时		223	167	84		—	— —
	显示结束时		79	59	30		—	— —

备注

- 1) *: 上面的一行表示当使用 A38B/A1S38B 和扩展基板时的处理时间。
 中间一行表示使用 A38HB/A1S38HB 时的处理时间。
 下面的一行表示当 Q312B 用来执行在插槽 0 中的 QJ71C24 指令的处理时间。
 FROM/TO 指令的处理时间会随插槽数和安装的模块数不同而有变化。
 (QnCPU/QnHCPU 中的处理时间也会随扩展基板的不同而有变化。)

指令	条件(软元件)	处理时间(μs)						
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
LEDC	当被显示时	559	420	210	—	—	—	—
	显示结束	413	310	155	—	—	—	—
LEDR	没有显示 → 没有显示	18	13	6.5	0.40	0.17	0.17	0.17
	LED 指令执行 → 没有显示	205	154	77	103	44	44	44
CHKST		15	11	5.5	5.8	2.5	2.5	2.5
CHK	触点 1 无错误	61	46	23	24	10	10	10
	触点 150 无错误	4232	3182	1591	1676	721	721	721
	触点 1 无错误	224	168	84	88	38	38	38
CHKCIR	步骤 10	15	11	5.5	5.8	2.5	2.5	2.5
SLT	所有的内部软元件	2399	1804	902	—	—	—	—
	文件寄存器 8K 数据点	7254	5454	2727	—	—	—	—
	SLT 执行完成	15	11	5.5	—	—	—	—
SLTR		1.1	0.8	0.4	—	—	—	—
STRA	开始	47	35	18	—	—	—	—
	STRA 指令完成	15	11	5.5	—	—	—	—
STRAR		1.1	0.8	0.4	—	—	—	—
PTRA		15	11	5.5	—	—	—	—
PTRAR		15	11	5.5	—	—	—	—
PTRAEXE	当进行操作时	1.6	1.2	0.6	—	—	—	—
PTRAEXEP	在进程中跟踪	169	127	64	—	—	—	—
BINDA	Ⓔ = 1	40	30	15	15	6.7	6.7	6.7
BINDAP	Ⓔ = - 32768	60	45	23	24	10	10	10
DBINDA	Ⓔ = 1	63	47	44	43	18	18	18
DBINDAP	Ⓔ = - 2147483648	217	163	82	86	37	37	37
BINHA	Ⓔ = 1	46	34	17	18	7.7	7.7	7.7
BINHAP	Ⓔ = FFFF _h	48	36	18	19	8.2	8.2	8.2
DBINHA	Ⓔ = 1	59	44	22	23	10	10	10
DBINHAP	Ⓔ = FFFFFFFF _h	62	46	23	24	10	10	10
BCDDA	Ⓔ = 1	58	43	22	23	9.8	9.8	9.8
BCDDAP	Ⓔ = 9999	54	40	20	21	8.9	8.9	8.9
DBCDDA	Ⓔ = 1	61	46	23	22	9.5	9.5	9.5
DBCDDAP	Ⓔ = 99999999	75	56	28	29	13	13	13
DABIN	Ⓔ = 1	133	100	50	57	25	25	25
DABINP	Ⓔ = - 32768	145	109	55	58	25	25	25
DDABIN	Ⓔ = 1	241	181	91	92	40	40	40
DDABINP	Ⓔ = - 2147483648	268	201	101	106	46	46	46
HABIN	Ⓔ = 1	32	24	12	13	5.8	5.8	5.8
HABINP	Ⓔ = FFFF _h	38	28	14	15	6.4	6.4	6.4
DHABIN	Ⓔ = 1	54	40	20	22	9.5	9.5	9.5
DHABINP	Ⓔ = FFFFFFFF _h	63	47	24	25	11	11	11

指令	条件(软元件)	处理时间(μs)						
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
DABCD	⑤ = 1	36	27	14		16	6.9	6.9
DABCDP	⑤ = 9999	42	31	16		17	7.2	7.2
DDABCD	⑤ = 1	63	47	24		25	11	11
DDABCDP	⑤ = 99999999	75	56	28		29	13	13
COMRD		36	27	14		40	17	17
COMRDP								
LEN	1 字节	48	36	18		18	8.0	8.0
LENP	96 字节	229	172	86		86	37	37
STR		132	99	50		53	23	23
STRP								
DSTR		285	214	107		123	53	53
DSTRP								
VAL		258	194	97		95	41	41
VALP								
DVAL		402	302	151		166	72	72
DVALP								
ESTR		1337	1005	503		564	243	243
ESTRP								
EVAL	小数点格式 所有 2 位	242	182	91		100	43	43
EVALP	指数格式 所有 6 位	306	230	115		127	55	55
ASC ⑤ ⑥ n	n = 1	164	123	62		64	28	28
ASCP ⑤ ⑥ n	n = 96	780	586	293		289	125	125
HEX ⑤ ⑥ n	n = 1	161	121	61		60	26	26
HEXP ⑤ ⑥ n	n = 96	826	621	311		343	148	148
RIGHT ⑤ ⑥ n	n = 1	131	98	49		49	21	21
RIGHTP ⑤ ⑥ n	n = 96	354	266	133		131	56	56
LEFT ⑤ ⑥ n	n = 1	129	97	49		50	21	21
LEFTP ⑤ ⑥ n	n = 96	354	266	133		131	56	56
MIDR		141	106	53		53	23	23
MIDRP								
MIDW		341	256	128		128	55	55
MIDWP								
INSTR	不匹配	156	117	59		58	25	25
NISTRP	匹配	开始	141	106	53	55	24	24
		最后	155	116	58	58	25	25

指令	条件(软元件)		处理时间(μs)					
			Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH QnPH
EMOD EMODP			1313	987	494		527	227
EREXP EREXPP			4423	3325	1663		1656	713
SIN	单精度		—	—	—	35	115	50
SINP	双精度		4921	3700	1850	—	1945	837
COS	单精度		—	—	—	35	122	53
COSP	双精度		6462	4858	2429	—	2618	1127
TAN	单精度		—	—	—	38	123	53
TANP	双精度		6515	4898	2449	—	2618	1127
√ASIN	单精度		—	—	—	44	111	48
ASINP	双精度		890	669	335	—	2491	1072
ACOS	单精度		—	—	—	44	115	49
ACOSP	双精度		801	602	301	—	2367	1019
ATAN	单精度		—	—	—	39	157	68
ATANP	双精度		7818	5878	2939	—	3140	1352
RAD	单精度		—	—	—	31	17	7.2
RADP	双精度		465	349	175	—	24	10
DEG	单精度		—	—	—	31	17	7.2
DEGP	双精度		492	369	185	—	23	9.9
SQR	单精度		—	—	—	39	28	12
SQRP	双精度		4520	3398	1699	—	1812	780
EXP EXPP	单精度	Ⓢ = -10	—	—	—	37	130	56
		Ⓢ = 1	—	—	—	37	130	56
	双精度	Ⓢ = -10	5871	4414	2207	—	—	1050
		Ⓢ = 1	5950	4474	2237	—	2400	1050
LOG LOGP	单精度	Ⓢ = 1	—	—	—	37	105	44
		Ⓢ = 10	—	—	—	37	115	49
	双精度	Ⓢ = 1	1191	896	448	—	470	205
		Ⓢ = 10	6839	5142	2571	—	2800	1200
RND RNDP			10	7.5	3.8		3.9	1.7
SRND SRNDP			8.8	6.6	3.3		3.5	1.5

指令	条件(软元件)	处理时间(μs)						
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
BSQR	Ⓔ = 0	16	12	6.0		6.2	2.7	2.7
BSQRP	Ⓔ = 9999	97	73	37		38	16	16
BDSQR	Ⓔ = 0	17	13	6.5		6.2	2.7	2.7
BDSQRP	Ⓔ = 99999999	88	66	33		38	16	16
BSIN		30	22	11		12	5.1	5.1
BSINP								
BCOS		32	24	12		12	5.2	5.2
BCOSP								
BTAN		30	22	11		12	5.2	5.2
BTANP								
BASIN		52	39	20		20	8.7	8.7
BASINP								
BACOS		53	40	20		21	9.0	9.0
BACOSP								
BATAN		56	42	21		22	9.6	9.6
BATANP								
LIMIT		24	18	9.0		10	4.3	4.3
LIMITP								
DLIMIT		28	21	11		11	4.7	4.7
DLIMITP								
BAND		24	18	9.0		9.8	4.2	4.2
BANDP								
DBAND		28	21	11		11	4.9	4.9
DBANDP								
ZONE		24	18	9.0		9.1	3.9	3.9
ZONEP								
DZONE		28	21	11		11	4.6	4.6
DZONEP								
RSET		19	14	7.0		6.8	2.9	2.9
RSETP								
QDRSET		322	242	121		205	88	88
QDRSETP								
QCDSET		218	164	82		147	63	63
QCDSETP								
DATERD		36	27	14		13	5.5	5.5
DATERDP								
DATEWR		42	31	16		15	6.4	6.4
DATEWRP								

指令	条件(软元件)	处理时间(μs)					
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH QnPH
DATE +	没有进位	60	45	23		13	5.4
DATE + P	有进位	60	45	23		13	5.4
DATE -	没有进位	59	44	22		12	5.2
DATE - P	有进位	60	45	23		12	5.2
SECOND SECONDP		27	20	10		10	4.5
HOUR HOURP		31	23	12		12	5.2
MSG	1 个字符	7.2	5.4	2.7		3.0	1.3
	32 个字符	7.4	5.6	2.8		3.0	1.3
PKEY	初始化时间	51	38	19		20	8.6
	无验收	48	36	18		19	8.2
PSTOP PSTOPP		122	92	46		79	34
POFF POFFP		120	90	45		79	34
PSCAN PSCANP		122	92	46		75	32
PLOW PLOWP		124	93	47		80	34
WDT WDT P		12	8.7	4.4		5.9	2.6
DUTY		1.6	1.2	0.6		9.3	4.0
ZRRDB ZRRDBP		19	14	6.9		7.9	3.4
ZRWRB ZRWRBP		21	16	7.8		9.4	4.0
ADRSET ADRSETP		13	9.3	4.7		4.9	2.1
KEY		43.4	32.4	16.2		17	7.3
ZPUSH ZPUSHP		27.6	20.6	10.3		11	4.7
ZPOP ZPOPP		12.7	9.5	4.8		5.1	2.2
EROMWR EROMWRP		62.6	46.7	23.4		—	—

指令	条件(软元件)	处理时间(μs)						
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
ZCOM		4296.6	3206.4	1603.2		691	289	289
READ		770.6	575.1	287.6		554	260	260
SREAD		858.9	641.0	320.5		588	278	278
WRITE		791.9	591.0	295.5		582	273	273
SWRITE		848.6	633.3	316.6		625	295	295
SEND		575.7	429.6	214.8		—	—	—
RECV		375.9	280.5	140.3		—	—	—
REQ		527.4	393.6	196.8		—	—	—
ZNFR		982.1	732.9	366.5		—	—	—
ZNTO		989.3	738.3	369.2		—	—	—
ZNRD	MELSECNET/10	598.6	446.7	223.4		—	—	—
	MELSECNET (11)	649.2	484.5	242.3		—	—	—
ZNWR	MELSECNET/10	614.3	458.4	229.2		—	—	—
	MELSECNET (11)	665.6	496.7	248.4		—	—	—
RFRP		590.9	441.0	220.5		—	—	—
RTOP		588.8	439.4	219.7		—	—	—

(4) QCPU 指令的处理时间(只包括 QCPU 指令)

(a) 功能版本 A 中可用的指令

指令	条件(软元件)		处理时间(μs)						
			Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
UNIRD			—	—	—	—	79	34	34
TRACE	开始		—	—	—	—	176	76	76
	STRA 执行结束		—	—	—	—	6.3	2.7	2.7
TRACER			—	—	—	—	19	8.2	8.2
FWRITE			—	—	—	—	84	36	36
FREAD			—	—	—	—	82	35	35
PLOADP			—	—	—	—	58	25	25
PUNLOADP			—	—	—	—	272	117	117
PSWAPP			—	—	—	—	308	133	133
RBMV	传送时间	1 个数据点	—	—	—	—	69	29	29
	使用 SRAM 的文件 R → 文件 R 的传送	1000 个数据点	—	—	—	—	580	308	308

(b) 功能版本 B 中可用的指令

指令	条件(软元件)		处理时间(μs)						
			Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
COM *1	带 CPU 共享存储区的自动刷新	刷新范围: 2k 字 (0.5k 字被平均分配到所有站点)	—	—	—	—	720	660	660
		刷新范围: 4k 字 (1k 字被平均分配到所有站点)	—	—	—	—	860	730	730
	不带 CPU 共享存储区的自动刷新	—	—	—	—	—	43	20	20
FROM *1	从其它站点的 CPU 共享存储区内读数据	n3 = 0	—	—	—	—	59	29	29
		n3 = 1000	—	—	—	—	530	500	500
	从智能功能模块的缓冲存储区内读数据 *2	n3 = 1	主基板单元	—	—	—	51	24	24
			扩展基板单元	—	—	—	54	27	27
		n3 = 1000	主基板单元	—	—	—	540	480	480
			扩展基板单元	—	—	—	1100	1050	1050
S. T0	向主站点的 CPU 共享存储区内写数据	n2 = 1	—	—	—	—	74	33	33
		n2 = 256	—	—	—	—	126	54	54

*1: 在一个多 PLC 系统中, 如果指令的处理和其它站点的处理重叠, 则处理时间增加下列时间中的最大值。

对于只包含主基板单元的系统
 (指令处理时间的增加) = $0.54 \times (\text{被处理的点数}) \times (\text{其它站点数}) (\mu\text{s})$

对于包含扩展基板单元的系统
 (指令处理时间的增加) = $1.30 \times (\text{被处理的点数}) \times (\text{其它站点数}) (\mu\text{s})$

*2: 在一个多 CPU 系统中, 处于自站控制下的智能功能模块的指令处理时间和处于其它站点控制下的智能功能模块的处理时间相同。

(5) 冗余系统指令(只用于 Q4ARCPU)

指令	条件(软元件)	处理时间(μs)						
		Q2A, Q2AS	Q3A	Q4A, Q2ASH	Q4AR	Qn	QnH	QnPH
S. STMODE		—	—	—	22.6	—	—	—
S. CGMODE		—	—	—	19.4	—	—	—
S. TRUCK		—	—	—	43.7	—	—	—
S. SPREF	缓冲存储区首地址 = 0 当使用 A68AD 时	—	—	—	407.1	—	—	—
	缓冲存储区首地址 = 0 当使用 A62DA 时	—	—	—	331.4	—	—	—

附录 2 CPU 之间的性能比较

2.1 Q/QnACPU 和 AnNCPU、AnACPU、以及 AnUCPU 之间的比较

2.1.1 可用的软元件

表 2.1 软元件比较

软元件名称		QCPU		QnACPU	AnUCPU	AnACPU	AnNCPU
输入/输出点数	Q00J: 256 点 Q00: 1024 点 Q01: 1024 点	Q02	4096 点	—	—	—	A1N: 256 点
		Q02H		Q2A: 512 点	A2U: 512 点	A2A: 512 点	A2N: 512 点
		Q06H		Q2A-S1: 1024 点	A2U-S1: 1024 点	A2A-S1: 1024 点	A2N-S1: 1024 点
		Q12H		Q3A: 2048 点	A3U: 2048 点	A3A: 2048 点	A3N: 2048 点
		Q25H		Q4A: 4096 点	A4U: 4096 点	—	—
		Q12PHCPU					
		Q25PHCPU					
内部继电器		8192 点 * 1			总共 8192 点	总共 8192 点	总共 2048 点
锁存继电器		2048 点 * 1	8192 点 * 1				
步进继电器		—					
顺序指令	—	—					
电器	SFC	2048 点	8192 点		—	—	—
报警器	1024 点 * 1	2048 点 * 1		2048 点	2048 点	256 点	
边沿触发继电器	1024 点 * 1	2048 点 * 1		—	—	—	
链接继电器	2048 点 * 1	8192 点 * 1		8192 点	4096 点	1024 点	
用于链接的特殊继电器	1024 点	2048 点		56 点	56 点	56 点	
定时器	512 点 * 1	2048 点 * 1		总共 2048 点	总共 2048 点	总共 256 点	
累计定时器	0 点 * 1						
计数器	512 点 * 1	1024 点 * 1		1024 点	1024 点	256 点	
数据寄存器	11136 点 * 1	12288 点 * 1		8192 点	6144 点	1024 点	
链接寄存器	2048 点 * 1	8192 点 * 1		8192 点	4096 点	1024 点	
用于链接的特殊寄存器	1024 点	2048 点		56 点	56 点	56 点	
函数输入		16 点(FX0 到 FXF)			—	—	—
函数输出		16 点(FY0 到 FYF)			—	—	—
特殊继电器		2048 点			256 点	256 点	256 点
函数寄存器		5 点(FD0 到 FD4)			—	—	—
特殊寄存器		2048 点			256 点	256 点	256 点
链接直接软元件		由 J 指定 <input type="checkbox"/> \ <input type="checkbox"/>			—	—	—
特殊直接软元件		由 U 指定 <input type="checkbox"/> \ G <input type="checkbox"/>			—	—	—
变址寄存器	Z	10 点 (Z0 到 Z9)	16 点(Z0 到 Z15)		7 点 (Z, Z1 到 Z6)	7 点 (Z, Z1 到 Z6)	1 点(Z)
	V * 2	—	—		7 点 (V, V1 到 V6)	7 点 (V, V1 到 V6)	1 点(V)
文件寄存器		32768 点 (R0 到 R32767)	32767 点/块 (R0 到 R32767)		8192 点/块 (R0 到 R8191)	8192 点/块 (R0 到 R8191)	8192 点/块 (R0 到 R8191)
累加器 * 3		—			2 点	2 点	2 点
嵌套		15 点			8 点	8 点	8 点
指针		300 点	4096 点		256 点	256 点	256 点
中断指针		128 点	256 点	48 点	32 点	32 点	32 点
SFC 块		—	320 点		—	—	—
SFC 转移软元件		—	512 点		—	—	—
十进制常数		K - 2147483648 到 K2147483647			K - 2147483648 到 K2147483647	K - 2147483648 到 K2147483647	K - 2147483648 到 K2147483647
十六进制常数		H0 到 HFFFFFFF			H0 到 HFFFFFFF	H0 到 HFFFFFFF	H0 到 HFFFFFFF
实数常数		—	E ± 1.17549-38 到 E ± 3.40282+38		—	—	—
字符串		"QnACPU", "ABCD" * 4			—	—	—

* 1: 软元件的点数可以在参数上修改。

* 2: QCPU/QnACPU 使用 V 作为边沿继电器。

* 3: 在 AnNCPU、AnACPU 和 AnUCPU 中使用的累加器指令和在 Q/QnACPU 中使用的格式不同。

* 4: 只能在 Q00JCPU, Q00CPU 和 Q01CPU 中被 \$MOV 指令使用。

2.1.2 I/O 控制模式

表 2.2 I/O 控制模式

I/O 控制模式			QCPU	QnACPU	AnUCPU	AnACPU	AnNCPU
刷新模式	直接 I/O 模式	部分刷新指令	○	○	○	○	○*2
		专用指令*1	○	○	○	○	○
		直接访问输入	—	—	○	○	—
		直接访问输出	○	○	—	—	—
	直接模式		—	—	—	—	○*2

表中的符号 ○：可用，—：不可用
 *1 DOUT、DSET 和 SRST 指令是直接输出专用指令。这里没有用于直接输出的专用指令。
 *2 在刷新模式和直接模式之间的切换是由 AnNCPU DIP 开关完成的。

2.1.3 可以被指令使用的数据

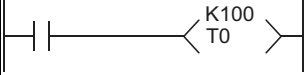
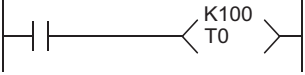
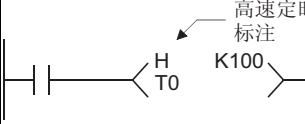
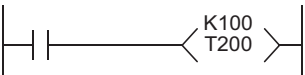
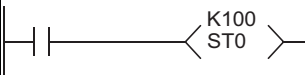
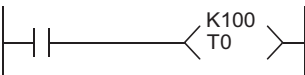
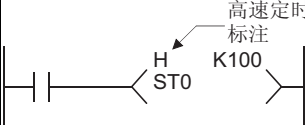
表 2.3 可以被指令使用的数据

设定数据		QCPU	QnACPU	AnUCPU	AnACPU	AnNCPU
位数据	位软元件	○	○	○	○	○
	字软元件	○ (需要位指定)	—	—	—	—
字数据	位软元件	○ (需要位指定)	○ (需要位指定)	○ (需要位指定)	○ (需要位指定)	○ (需要位指定)
	字软元件	○	○	○	○	○
双精度数据	位软元件	○ (需要位指定)	○ (需要位指定)	○ (需要位指定)	○ (需要位指定)	○ (需要位指定)
	字软元件	○	○	○	○	○
实数数据		○*1	○	○	—	—
字符串数据		○*2	—	—	—	—

表中的符号 ○:可用，—:不可用
 *1 不可用于 Q00J/Q00/Q01CPU。
 *2 只能用于 Q00J/Q00/Q01CPU 的\$MOV 指令。

2.1.4 定时器比较

表 2.4 定时器比较

功能		QCPU/QnACPU	AnUCPU	AnACPU	AnNCPU
低速定时器	测量单元	● 100ms(默认值) 可以在 10 到 100ms 范围内改变(参数设定)	● 固定在 100 ms		
	指定方法				
高速定时器	测量单元	● 10ms(默认值) 可以在 1 到 100ms 范围内改变(参数设定)	● 固定在 10ms		
	指定方法	 * 高速定时器设置: 由顺序程序执行	 * 高速定时器设置: 在参数处执行		
累计定时器	测量单元	● 和低速定时器一样的测量单元	● 固定在 10ms		
	指定方法				
高速累计定时器	测量单元	● 和高速定时器一样的测量单元	● 没有		
	指定方法	 * 高速定时器设置: 由顺序程序执行			
设定值的设定范围		● 1 到 32767	● 1 到 32767		
对设定值 0 的处理		● 即刻 ON	● 没有最大值(没有时间溢出)		
索引修改	触点	● 允许(只有 Z0 和 Z1 可用)	● 可以	● 不可以	
	线圈	● 允许(只有 Z0 和 Z1 可用)	● 不可以	● 不可以	
	设定值	● 不可以	● 不可以	● 不可以	
	当前值	● 允许(Z0 到 Z15 可用)	● 可以	● 可以	
当前值的更新处理		● 当 OUT Tn 指令被执行时		● 当 END 处理被执行时	
触点 ON/OFF 处理					

(1) 使用定时器时的注意事项

Q/QnACPU 更新定时器的当前值，并在 OUT T [] 指令被执行时将它们的触点切换为 ON/OFF。因此，如果当定时器线圈变为 ON 时“当前值 ≥ 设定值”，此定时器的触点变为 ON。

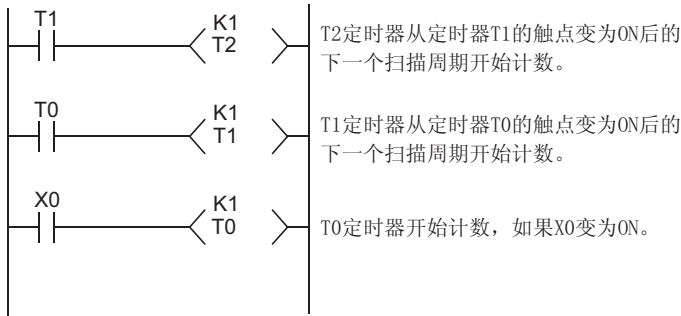
当创建一个程序，在此程序中一个定时器触点的操作将触发其它定时器的操作时，按照定时器的操作顺序来创建程序 — 首先创建较后操作的定时器的程序。

在下列的情况下，如果程序按照定时器操作的顺序创建，则所有的定时器将在同一扫描时刻变为 ON。

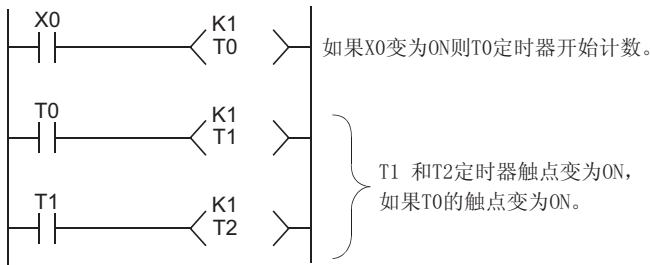
- 对于高速定时器，如果设定值小于一个扫描时间。
- 对于低速定时器，如果“1”被设定。

例子

- 对于定时器从 T0 到 T2，按照较后操作的定时器在先的顺序创建程序。



- 对于定时器从 T0 到 T2，按照定时器的执行顺序创建程序。



2.1.5 计数器比较

表 2.5 计数器比较

功能		QCPU/QnACPU	AnUCPU	AnACPU	AnNCPU
指定方法					
索引修改	触点	● 允许(只有 Z0 和 Z1 可用)	● 可以		● 不可以
	线圈	● 允许(只有 Z0 和 Z1 可用)	● 可以		● 不可以
	设定值	● 不可以	● 不可以		● 不可以
	当前值	● 允许(Z0 到 Z15 可用) *	● 可以		● 可以
对当前值进行更新处理		● 当 OUT Tn 指令被执行时		● 当 END 处理被执行时	
触点的 ON/OFF 处理					

*: Q00J/Q00/Q01CPU 可以使用 Z0 到 Z9。

2.1.6 显示指令比较

表 2.6 显示指令比较

指令	QCPU/QnACPU	AnUCPU	AnACPU	AnNCPU
PR *	● 当 SM701 变为 OFF: 连续输出直到碰到 00h ● 当 SM701 变为 ON: 16 字符输出	● 当 M9049 变为 OFF: 连续输出直到碰到 00h ● 当 M9049 变为 ON: 16 字符输出		
PRC *	● 当 SM701 变为 OFF: 32 字符注释输出 ● 当 SM701 变为 ON: 高 16 字符输出	16 字符注释输出		

*: 不可以用于 Q00J/Q00/Q01CPU。

2.1.7 改变定义格式指令(不包括用于 AnACPU 和 AnUCPU 的专用指令)

由于 Q/QnACPU 没有累加器(A0, A1), AnUCPU、AnACPU 和 AnNCPU 这些使用累加器的指令的格式发生了改变。

表 2.7 改变表达方式的指令


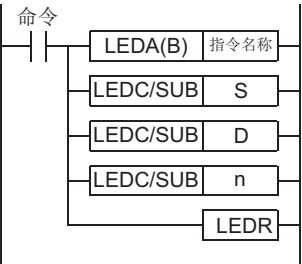
功能	QCPU/QnACPU		AnUCPU/AnACPU/AnNCPU	
	指令格式	备注	指令格式	备注
16 位循环右移		● D: 循环移动的数据		● 循环移动的数据在 A0 处设定
		● D: 循环移动的数据 ● 进位标志使用 SM700		● 循环移动的数据在 A0 处设定 ● 进位标志使用 M9012
16 位循环左移		● D: 循环移动的数据		● 循环移动的数据在 A0 处设定
		● D: 循环移动的数据 ● 进位标志使用 SM700		● 循环移动的数据在 A0 处设定 ● 进位标志使用 M9012
32 位循环右移		● D: 循环移动的数据		● 循环移动的数据在 A0 和 A1 处设定
		● D: 循环移动的数据 ● 进位标志使用 SM700		● 循环移动的数据在 A0 和 A1 处设定 ● 进位标志使用 M9012
32 位循环左移		● D: 循环移动的数据		● 循环移动的数据在 A0 和 A1 处设定
		● D: 循环移动的数据 ● 进位标志使用 SM700		● 循环移动的数据在 A0 和 A1 处设定 ● 进位标志使用 M9012
16 位数据查找		● 搜索结果存储在 D 和 D+1 软元件中		● 搜索结果存储在 A0 和 A1
32 位数据查找		● 搜索结果存储在 D 和 D+1 软元件中		● 搜索结果存储在 A0 和 A1
16 位数据位检查		● 检查结果存储在 D 软元件中		● 检查结果存储在 A0
16 位数据位检查		● 检查结果存储在 D 软元件中		● 检查结果存储在 A0
分区刷新		● 新增的专用指令		● 只有在 M9052 为 ON 时
8 字符 ASCII 转换				
进位标志置位		● 无专用指令		
进位标志复位		● 无专用指令		
跳转到 END 指令		● 新增的专用指令		● P255: END 指令指定
CHK 指令*		● 新增的专用指令		

*: 不能用于 Q00J/Q00/Q01CPU。

2.1.8 AnACPU 和 AnUCPU 专用指令

- (1) 专用指令的表达方法
- 在 Q/QnACPU 中，带 AnACPU 或 AnUCPU 的 LEDA、LEDB、LEDC、SUB 和 LEDR 指令的专用指令已经被改变格式，以便和基本指令和应用指令保持一致。
- 那些由于在 QCPU/QnACPU 中没有相应指令而不能被转换的指令，被转换成 OUT SM1255/OUT SM999（用于 Q00J/Q00/Q01CPU）。
- 那些已经被转换成 OUT SM1255/OUT SM999 的指令应该被其它指令替换掉或者被删除。

表 2.8 专用指令的表达方法

QCPU/QnACPU	AnUCPU/AnACPU
	
	*: S, D, 和 n 表示被指令使用的数据。

- (2) 被改名的的专用指令
- 用于 AnUCPU 或 AnACPU 的专用指令，它们和基本指令或应用指令使用同样的指令名字，它们在 Q/QnACPU 中会被改名。

表 2.9 被改名的专用指令

功能	QCPU/QnACPU	AnUCPU/AnACPU
浮动小数点数据加	E +	ADD
浮动小数点数据减	E -	SUB
浮动小数点数据乘	E *	MUL
浮动小数点数据除	E /	DIV
数据分解	NDIS	DIS
数据组合	NUNI	UNI
更新检查模式	CHKCIR, CHKEND	CHK, CHKEND

2.1.9 只能在一般模式中编程的指令

- 以下指令只能用于在 QnACPU 的一般模式中编程：
(当程序创建后，程序被编译并被转换成实际的顺序程序。)
- CHKCIR 到 CHKEND(更新检查指令的检查模式)
 - IX 到 IXEND(用于整个梯形图的索引修改)
 - IXDEV, IXSET

附录 3 特殊继电器列表

附录 3.1 基本型 QCPU 的特殊继电器列表

特殊继电器，SM，是应用固化在 PLC 中的内部继电器。
 由于这个原因，它们不能和普通的内部继电器一样被顺控程序使用。
 然而，如果需要，它们可以变为 ON 或 OFF 以控制 CPU 模块和远程 I/O 模块。

表中标题的含义如下所示。

项目	项目的功能
号码	● 表示特殊继电器的号码
名称	● 表示特殊继电器的名称
含义	● 表示特殊继电器的种类
解释	● 包含特殊继电器内容的详细信息。
设置者(当被设置时)	● 表示继电器是被系统设置还是被用户设置；并且，如果是被系统设置，当设置被执行时。 <设置者> S : 由系统设置 U : 由用户设置（在外围软元件的顺序程序或者测试操作中） S/U : 由系统和用户设置 <当被设置时> → 只有当设置由系统执行时才被指示 每个 END : 在每条 END 处理过程中被设置 初始化 : 在初始化处理时被设置 （当系统电源变为 ON，或者当从 STOP 变成 RUN 时） 状态改变 : 只有在状态有改变时才被设置 错误 : 只有在错误产生时才被设置 指令执行 : 在指令被执行时被设置 请求 : 只有在有用户请求时才被设置 （通过 SM，等）
相应的 CPU	● 表示相应的 CPU 模块的类型名称 ○ : 可以被应用到所有类型的 CPU 模块 每个 CPU 模块的类型名称 : 只能被应用到指定的 CPU 模块

对于下列项目的详细信息，参见下列手册：

- 网络 → ● 网络系统参考手册

特殊继电器列表

(1) 诊断信息


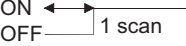




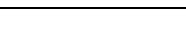
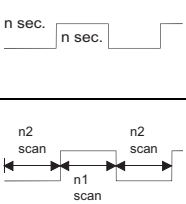
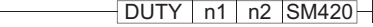
号码	名称	含义	解释	设置者 (当被设置时)	相应的 CPU
SM0	诊断错误	OFF : 无错误 ON : 错误	● 如果作为诊断结果有错误发生, 变为 ON。 (包括当一个报警器变为 ON) ● 如果以后条件重新变为正常, 保持 ON。	S (错误)	○
SM1	自诊断错误	OFF : 无自诊断错误 ON : 自诊断	● 如果作为诊断结果有错误发生, 变为 ON。 (不包含当一个报警器变为 ON) ● 如果以后条件重新变为正常, 保持 ON。	S (错误)	
SM5	错误公共信息	OFF : 无错误公共信息 ON : 错误公共信息	● 当 SM0 为 ON 时, 如果有错误公共信息, 变为 ON。	S (错误)	
SM16	错误单独信息	OFF : 无错误公共信息 ON : 错误公共信息	● 当 SM0 为 ON 时, 如果有错误单独信息, 变为 ON。	S (错误)	
SM50	错误复位	OFF →ON : 错误复位	● 执行错误复位操作 参见章节 11.3 以获得进一步信息	U	
SM51	电池电压低保持	OFF : 正常 ON : 电池电压低	● 如果 CPU 模块上的电池电压低于额定值, 变为 ON。 ● 即使以后电池电压回到正常, 还保持为 ON。 ● 和 ERR, LED, 保持同步	S (错误)	
SM52	电池电压低	OFF : 正常 ON : 电池电压低	● 和 SM51 一样, 但是当电池电压回到正常后最终变为 OFF	S (错误)	
SM53	AC/DC DOWN 检测	OFF : AC/DC DOWN 不被检测 ON : AC/DC DOWN 被检测	● 如果在使用 AC 电源供电模块时发生 20ms 内的瞬时电源故障, 变为 ON。 ● 当电源切换到 OFF 时复位, 然后为 ON。	S (错误)	
			● 如果在使用 DC 电源供电模块时发生 10ms 内的瞬时电源故障, 变为 ON。 ● 当电源切换到 OFF 时复位, 然后为 ON。		
SM56	操作错误	OFF : 正常 ON : 操作错误	● 当产生一个操作错误时, 变为 ON。 ● 如果以后条件重新变为正常, 保持 ON。	S (错误)	
SM60	烧坏保险丝检测	OFF : 正常 ON : 模块有烧坏的保险丝	● 如果至少有一个输出模块的保险丝被烧坏, 变为 ON。 ● 如果以后条件重新变为正常, 保持 ON。	S (错误)	
SM61	I/O 模块确认错误	OFF : 正常 ON : 错误	● 如果 I/O 模块和在上电时指定的状态不同, 变为 ON。 ● 如果以后条件重新变为正常, 保持为 ON。	S (错误)	
SM62	报警器检测	OFF : 没有被检测 ON : 被检测	● 如果正好有一个报警器 (F) 变为 ON, 变为 ON 。	S (指令执行)	
SM100	串行通信功能使用标识	OFF : 串行通信功能没有被使用 ON : 串行通信功能被使用	● 将是否使用串行通信功能的设置存储到串行通信设置参数里	S (电源上电或复位)	
SM101	通信协议状态标识	OFF : GX Developer ON : MC 协议通信软元件	● 存储通过 RS-232 接口通信的软元件是 GX Developer 还是 MC 协议通信软元件。	S (RS232 通信)	
SM110	协议错误	OFF : 正常 ON : 不正常	● 当在串行通信功能中一个不正常的协议被用于通信, 变为 ON。 ● 如果以后条件重新变为正常, 保持 ON。	S (错误)	
SM111	通信状态	OFF : 正常 ON : 不正常	● 当被用于通信的模式和串行通信功能中设置的模式不同时, 变为 ON 。 ● 如果以后条件重新变为正常, 保持 ON。	S (错误)	
SM112	错误信息清除	ON : 已被清除	● 当存储在 SM110, SM111, SD110 和 SD111 中的错误代码被清除掉。(当从 OFF 变为 ON 时被激活), 变为 ON。	U	
SM113	超限错误	OFF : 正常 ON : 不正常	● 当在串行通信错误中发生超限错误时, 变为 ON。	S (错误)	
SM114	奇偶校验错误	OFF : 正常 ON : 不正常	● 当在串行通信错误中发生奇偶校验错误时, 变为 ON。	S (错误)	
SM115	帧错误	OFF : 正常 ON : 不正常	● 当在串行通信错误中发生帧错误时, 变为 ON。	S (错误)	

(2) 系统信息

号码	名称	含义	解释	设置者 (当被设置时)	可应用的 CPU
SM203	STOP 触点	STOP 状态	● 在 STOP 状态变为 ON。	S (状态 改变)	○
SM204	PAUSE 触点	PAUSE 状态	● 在 PAUSE 状态变为 ON。	S (状态改变)	
SM206	PAUSE 使能线圈	OFF : PAUSE 无效 ON : PAUSE 使能	● 如果当远程 PAUSE 触点变为 ON 时此继电器为 ON。进入 PAUSE 状态。	U	
	软元件测试请求接受状态	OFF : 软元件测试没有被执行 ON : 软元件测试被执行	● 当在 GX Developer 上执行软元件测试模式时, 变为 ON。	S (请求)	
SM210	时钟数据设置请求	OFF : 忽略 ON : 设置请求	● 当此继电器从 OFF 变为 ON 时, 扫描周期发生变化的 END 指令执行完毕后, 存储在从 SD210 到 SD213 的时钟数据被写入到时钟软元件中。	U	
SM211	时钟数据错误	OFF : 无错误 ON : 错误	● 当在时钟数据(SD210 到 SD213)值中发生错误时变为 ON, 并在无错误被检测到时变为 OFF。	S (请求)	
SM213	时钟数据写请求	OFF : 忽略 ON : 读请求	● 当此继电器为 ON 时, 时钟数据从 SD213 以 BCD 数值读到 SD210 里面。	U	
SM240	第 1 CPU 复位标识	OFF : 第 1 CPU 复位取消 ON : 第 1 CPU 重新设置	● 当对第 1 CPU 的复位被取消时, 变为 OFF。 ● 当第 1 CPU 被重新设置时(包含 CPU 被从基板中移走的情况), 变为 ON。 其它 CPU 也进入复位状态。	S (状态改变)	○ 序列号 04122 或更大
SM241	第 2 CPU 复位标识	OFF : 第 2 CPU 复位取消 ON : 第 2 CPU 重新设置	● 当对第 2 CPU 的复位被取消时, 变为 OFF。 ● 当第 2 CPU 被重新设置时(包含 CPU 被从基板中移走的情况), 变为 ON。 其它 CPU 的结果为“MULTI CPU DOWN”(错误 代码: 7000)。		
SM242	第 3 CPU 复位标识	OFF : 第 3 CPU 复位取消 ON : 第 3 CPU 重新设置	● 当对第 3 CPU 的复位被取消时, 变为 OFF。 ● 当第 3 CPU 重新设置时(包含 CPU 被从基板中移走的情况), 变为 ON。 其它 CPU 的结果为“MULTI CPU DOWN”(错误代码: 7000)。		
SM244	第 1 CPU 错误标识	OFF : 第 1 CPU 正常 ON : 第 1 CPU 处于停止错误	● 当第 1 CPU 为正常时(包含一个连续错误), 变为 OFF。 ● 当第 1 CPU 处于一个停止错误时, 变为 ON。		
SM245	第 2 CPU 错误标识	OFF : 第 2 CPU 正常 ON : 第 2 CPU 处于停止错误	● 当第 2 CPU 为正常(包含一个连续错误)时 , 变为 OFF。 ● 当第 2 CPU 处于一个停止错误时, 变为 ON。		
SM246	第 3 CPU 错误标识	OFF : 第 3 CPU 正常 ON : 第 3 CPU 处于停止错误	● 当第 3 CPU 为正常(包含一个连续错误)时 , 变为 OFF。 ● 当第 3 CPU 处于一个停止错误时, 变为 ON。		
SM315	通信预定时间延迟有效/无效标识	OFF : 无延迟 ON : 有延迟	● 当预定的通信处理时间在 SD315 中设置时, 此标识被使能。 ● 变为 ON 以按照设置在 SD315 中的时间来延迟 END 的处理, 以保证通信处理的执行。(扫描时间增加了设置在 SD315 中的时间) ● 当没有通信处理时, 变为 OFF 以执行 END 处理而不按照设置在 SD315 中的时间去延迟(默认为 OFF)。	U	○
SM320	存在/不存在 SFC 程序	OFF : SFC 程序不存在 ON : SFC 程序存在	● 当有 SFC 程序被登记, 变为 ON。 ● 当 SFC 程序没有被登记, 变为 OFF。	S (初始化)	○ 序列号 04122 或更大
SM321	启动/停止 SFC 程序	OFF : SFC 程序不执行(停止) ON : SFC 程序执行(开始)	● 和 SM320 一样的值被设置为初始值。 (自动变为 ON, 当 SFC 程序存在时。) ● 当此继电器从 ON 变为 OFF 时, 此 SFC 程序的执行被停止。 ● 当此继电器从 OFF 变为 ON 时, 此 SFC 程序被恢复。	S (初始化) U	
SM322	SFC 程序的开始状态	OFF : 初始化开始 ON : 连续开始	● 在 PLC 参数对话框的 SFC 设置中 SFC 程序启动模式被设置为初始值。 在初始化开始处: OFF 在连续开始处: ON	S (初始化) U	
SM323	存在/不存在所有块的连续转换	OFF : 无连续转换 ON : 有连续转换	在 SFC 数据软元件的“连续转换位”没有被设置的地方设置块的存在/不存在连连转换。	U	
SM324	连续转换禁止标识	OFF : 当转换被执行时 ON : 当转换不被执行时	● 在连续转换模式操作中或连续转换中变为 OFF, 当连续转换不被执行时变为 ON。 ● 常 ON, 在没有连续转换模式的操作中	S (状态改变)	

号码	名称	含义	解释	设置者 (当被设置时)	可用的 CPU
SM325	块停止时间输出模式	OFF : OFF ON : 保持	在块停止时刻选择活动步的线圈输出是否保持。 ● 作为初始值, 当线圈输出为 OFF 时参数中的块停止输出模式为 OFF, 并在线圈输出被保持时变为 ON。 ● 当此继电器为 OFF, 线圈输出都变为 OFF。 ● 当此继电器变为 ON, 线圈输出被保持。	S (初始化) U	○ 序列号为 04122 或更大
SM326	SFC 软元件清除模式	OFF : 软元件被清除 ON : 软元件保持	在从 STOP 到写程序到 RUN 的切换时刻, 选择软元件状态。 (除步进继电器外的所有软元件)	U	
SM327	结束步执行时的输出	OFF : 保持步输出 OFF ON : 保持步输出保持	● 当此继电器变为 OFF 时, 当转换建立时被保持的步 (SC, SE, ST) 的线圈输出在结束步到达时变为 OFF。	U	
SM328	当结束步到达时进入清除处理模式	OFF : 清除处理被执行 ON : 清除处理不被执行	在结束步到达时, 如果块中存在的是活动步而不是其它被保持的步, 选择是否执行清除处理。 ● 当此继电器变为 OFF 时, 所有的活动步被强制结束以结束整个块。 ● 当此继电器为 ON 时, 此块的执行被继续。 ● 当结束步到达时, 如果活动步而不是其它被保持的步不存在, 结束被保持的步以结束整个块。	U	

(3) 系统时钟/计数器

号码	名称	含义	解释	设置者 (当被设置时)	可用的 CPU
SM400	常 ON	ON _____ OFF	● 正常为 ON	S (每个 END 处理)	○
SM401	常 OFF	ON _____ OFF _____	● 正常为 OFF	S (每个 END 处理)	
SM402	只在 RUN 后 ON 一个	ON  OFF	● 在 RUN 后, 只 ON 一个扫描周期。 ● 此连接只能用于扫描执行类型程序。	S (每个 END 处理)	
SM403	在 RUN 后, 只 OFF 一个扫描周期	ON  OFF	● 在 RUN 后, 只 OFF 一个扫描周期。 ● 此连接只能用于扫描执行类型程序。	S (每个 END 处理)	
SM410	0.1 秒时钟		● 在每个指定的时间间隔内在 ON 和 OFF 之间重复改变。 ● 当电源变为 OFF, 或执行复位操作时, 从 OFF 开始启动。 * 注意, 在程序的执行过程中, 当指定的时间用完时 ON-OFF 状态改变	S (状态改变)	
SM411	0.2 秒时钟				
SM412	1 秒时钟				
SM413	2 秒时钟				
SM414	2n 秒时钟		● 此继电器按照在 SD414 中设置的时间间隔(单位:s)在 ON 和 OFF 之间切换。 ● 当 PLC 电源为 ON 或 CPU 模块被复位时从 OFF 开始启动。 ● 注意, 当指定的时间到达时, 即使在程序的执行过程中, ON/OFF 状态也会改变。	S (状态改变)	
SM420	0 号用户定时时钟		● 在固定的时间间隔上, 继电器重复 ON/OFF 切换。 ● 当电源变为 ON, 或执行复位时, 从 OFF 开始启动。 ● ON/OFF 间隔由 DUTY 指令设置。	S (每个 END 处理)	
SM421	1 号用户定时时钟				
SM422	2 号用户定时时钟				
SM423	3 号用户定时时钟				
SM424	4 号用户定时时钟				
					

(4) I/O 刷新

号码	名称	含义	解释	设置者 (当被设置时)	相应的 CPU
SM580	程序到程序 I/O 刷新	OFF : 不刷新 ON : 刷新	<ul style="list-style-type: none"> 当此特殊继电器变为 ON 时, 在第一个程序执行后执行 I/O 刷新, 然后第二个程序开始执行 当一个顺序程序和一个 SFC 程序被执行时, 顺序程序被执行, I/O 刷新被执行, 然后是 SFC 程序被执行。 	U	○ 序列号 04122 或更大

(5) 存储卡

号码	名称	含义	解释	设置者 (当被设置时)	可用的 CPU
SM620	驱动器 3/4 可用标识	OFF : 不可用 ON : 可用	<ul style="list-style-type: none"> 常 ON 	S (初始化)	○
SM621	驱动器 3/4 保护标识	OFF : 无保护 ON : 保护	<ul style="list-style-type: none"> 常 OFF 	S (初始化)	
SM622	驱动器 3 标识	OFF : 无驱动器 3 ON : 驱动器 3 存在	<ul style="list-style-type: none"> 常 ON 	S (初始化)	
SM623	驱动器 4 标识	OFF : 无驱动器 4 ON : 驱动器 4 存在	<ul style="list-style-type: none"> 常 ON 	S (初始化)	
SM640	文件寄存器使用	OFF : 文件寄存器未被使用 ON : 文件寄存器正被使用	<ul style="list-style-type: none"> 当文件寄存器处于使用 (只对 Q00CPU, Q01CPU) 时, 变为 ON。 	S (状态改变)	
SM660	引导操作	OFF : 程序存储区执行 ON : 引导操作正被处理	<ul style="list-style-type: none"> 当引导操作正被处理时, 变为 ON。 	S (状态改变)	

(6) 和特殊继电器相关的指令

号码	名称	含义	解释	设置者 (当被设置时)	可用的 CPU
SM700	进位标识	OFF : 进位 OFF ON : 进位 ON	<ul style="list-style-type: none"> 进位标识, 用于应用指令 	S (指令执行)	○
SM702	搜索方法	OFF : 搜索下一个 ON : 2-部分搜索	<ul style="list-style-type: none"> 被搜索指令使用的指定方法。 数据必须按照 2 部分搜索来排列。 	U	
SM703	排序	OFF : 上升序列 ON : 下降序列	<ul style="list-style-type: none"> 排序指令用于指定数据是按照升序排列还是按照降序排列。 	U	
SM704	块比较	OFF : 没有匹配发现 ON : 所有的都匹配	<ul style="list-style-type: none"> 当 BKCOMP 指令的所有数据条件都满足时, 变为 ON。 	S (指令执行)	
SM715	EI 标识	0 : 处于 DI 1 : 处于 EI	<ul style="list-style-type: none"> 当 EI 指令正被执行时, 为 ON。 	S (指令执行)	
SM721	文件正被访问	OFF : 文件没有被访问 ON : 文件正被访问	<ul style="list-style-type: none"> 当有文件被 S.FWRITE, S.FREAD, COMRD, PRC, 或 LEDC 指令访问时, 切换为 ON。 	S (状态改变)	
SM722	BIN/DBIN 指令错误禁止标识	OFF : 执行错误检测 ON : 错误检测不执行	<ul style="list-style-type: none"> 当“OPERATION ERROR”因 BIN 或 DBIN 指令而被抑制, 变为 ON。 	U	○ 序列号 04122 或更大
SM774	PID 无扰处理 (用于精确的微分)	OFF : 匹配 ON : 无匹配	<ul style="list-style-type: none"> 在手动模式中指示设置值 (SV) 是否和过程值 (PV) 相匹配。 	U	
SM775	在 COM 指令执行过程中选择链接刷新处理	OFF : 执行链接刷新 ON : 无链接刷新被执行	<ul style="list-style-type: none"> 在 COM 指令的执行过程中, 如果只有一般数据处理被执行, 选择是否执行链接刷新处理。 	U	○
		OFF : 所有的刷新处理都被执行 ON : 在 SD778 中的刷新设置被执行	<ul style="list-style-type: none"> 当 COM 指令被执行时, 选择是所有的刷新处理还是在 SD778 中设置的刷新处理被执行。 	U	○ 序列号 04122 或更大
SM794	PID 无扰处理 (用于精确的微分)	OFF : 匹配 ON : 不匹配	<ul style="list-style-type: none"> 在手动模式中的指示设置值 (SV) 是否和过程值 (PV) 相匹配。 	U	○ 序列号 04122 或更大

附录 3.2 高性能型 QCPU/QnACPU 的特殊继电器列表

特殊继电器，SM，是应用固化在 PLC 中的内部继电器。
 由于这个原因，它们不能像普通的内部继电器那样被顺控程序使用。
 然而，如果需要它们可以切换成 ON 或 OFF 以控制 CPU 模块和远程 I/O 模块。

下表中标题的含义如下所示。

项目	项目功能
号码	● 表示特殊继电器的号码。
名称	● 表示特殊继电器的名称。
含义	● 表示特殊继电器的种类。
解释	● 包含关于特殊继电器内容的详细信息。
设置者(当被设置时)	<ul style="list-style-type: none"> ● 表示继电器是有系统设置还是由用户设置，并且，如果是由系统设置，当设置被执行时。 <设置者> S :由系统设置 U :由用户设置(在 GXDeveloper 上的顺序程序或测试操作中) S/U :由系统和用户共同设置 <当被设置时>→表示只由系统设置 每个 END :在每个 END 处理过程中设置 初始化 :只在初始化处理过程中设置 (当电源变为 ON, 或者当从 STOP 变为 RUN 时) 状态改变 :在有状态改变时设置 错误 :当有错误产生时设置 指令执行 :当指令被执行时设置 请求 :当有用户请求时设置(通过 SM, 等)
相应的 ACPU M9□□□	<ul style="list-style-type: none"> ● 表示特殊继电器 M9 □ □ □ 和 ACPU 相对应。 (当注释中发生变化时改变和符号) ● “新增”表示此项目是新近增加到高性能模块 QCPU/QnACPU。
相应的 CPU	<ul style="list-style-type: none"> ● 表示相应的 CPU 模块类型名称 ○+Rem:可以被应用到所有的 CPU 类型和 MELSECNET/H 远程 I/O 模块 ○:可以被应用到所有类型的 CPU QCPU:可以应用到高性能模块 QCPU QnA:可以被应用到 QnA 系列和 Q2ASCPU 系列 远程:可以应用到 MELSECNET/H 远程 I/O 模块 每个 CPU 模块类型名称:只能应用到指定的 CPU 模块(如 Q4AR, Q3A)

对于下列项目的详细信息，参见下列手册：

- 网络 → ● 用于和 Q 相应的 MELSECNET/H 网络系统参考手册
 (PLC 到 PLC 网络)
- 用于 Q 相应的 MELSECNET/H 网络系统参考手册
 (远程 I/O 网络)
- 用于 QnA/Q4ARMELSECNET/10 网络系统参考手册
- SFC → QCPU(Q 模式)/QnACPU 编程手册(SFC)

要点
(1)SM1200 到 SM1255 用于 QnACPU。 这些继电器对 QCPU 空缺。
(2)特殊继电器 SM1500 和其之后的专用于 Q4ARCPU。

特殊继电器列表

(1) 诊断信息

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM0	诊断错误	OFF : 无错误 ON : 有错误	<ul style="list-style-type: none"> 如果作为诊断结果有错误发生, 变为 ON。 (包括当一个报警器为 ON, 并且当有错误被 CHK 指令检测到) 如果条件重新变为正常, 保持 ON。 	S (有错误)	新增	○+Rem
SM1	自诊断错误	OFF : 无自诊断错误 ON : 自诊断	<ul style="list-style-type: none"> 如果作为诊断结果有错误发生 (不包括当一个报警器变为 ON 或当有错误被 CHK 指令检测到), 变为 ON。 如果条件重新变为正常, 保持 ON。 	S (有错误)	M9008	
SM5	错误公用信息	OFF : 无错误公用信息 ON : 有错误公用信息	<ul style="list-style-type: none"> 当 SM0 为 ON 时, 如果有错误公用信息则变为 ON。 	S (有错误)	新增	
SM16	错误单独信息	OFF : 无错误公用信息 ON : 有错误公用信息	<ul style="list-style-type: none"> 当 SM0 为 ON 时, 如果有错误单独信息则变为 ON。 	S (有错误)	新增	
SM50	错误复位	OFF → ON : 错误复位	<ul style="list-style-type: none"> 执行错误复位操作。 参见章节 11.3 以获得进一步信息。 	U	新增	
SM51	电池低电压闭锁	OFF : 正常 ON : 电池电压低	<ul style="list-style-type: none"> 如果 CPU 模块或存储卡上的电池电压低于额定值, 则变为 ON。 如果电池电压重新变为正常, 保持为 ON。 同步于 BAT. ALARM/BAT. LED。 	S (有错误)	M9007	○
SM52	电池电压低	OFF : 正常 ON : 电池电压低	<ul style="list-style-type: none"> 和 SM51 一样, 但是在电池电压重新变为正常后最终变为 OFF。 	S (有错误)	M9006	
SM53	AC/DCDOWN 检测	OFF : AC/DCDOWN 不被检测 ON : AC/DCDOWN 被检测	<ul style="list-style-type: none"> 如果在使用 AC 电源供电模块时有 20ms 内的瞬时电源故障发生, 变为 ON。 当电源切换到 OFF 时复位, 然后变为 ON。 如果在使用 DC 电源供电模块时有 10ms 内的瞬时电源故障发生, 变为 ON。 当电源切换到 OFF 时复位, 然后变为 ON。 如果在使用 DC 电源供电模块时有 1ms 内的瞬时电源故障发生, 变为 ON。 当电源切换到 OFF 时复位, 然后变为 ON。 	S (有错误)	M9005	QCPU
						QnA
SM54	MINI 链接错误	OFF : 正常 ON : 有错误	<ul style="list-style-type: none"> 如果 MINI (S3) 链接错误在安装上的一个 AJ71PT32 (S3) 模块上被检测到, 变为 ON。 如果条件重新变为正常, 保持 ON。 	S (有错误)	M9004	QnA
SM56	操作错误	OFF : 正常 ON : 操作错误	<ul style="list-style-type: none"> 当操作错误发生时为 ON。 如果条件重新变为正常保持 ON。 	S (有错误)	M9011	○
SM60	烧坏保险丝检测	OFF : 正常 ON : 模块有烧坏的保险丝	<ul style="list-style-type: none"> 如果至少有一个输出模块的保险丝被烧坏, 变为 ON。 如果条件重新变为正常, 保持 ON。 即使对远程 I/O 站点的输出模块, 烧坏保险丝的状态也被检测。 	S (有错误)	M9000	○+Rem
SM61	I/O 模块验证错误	OFF : 正常 ON : 有错误	<ul style="list-style-type: none"> 如果 I/O 模块和上电时指定的状态不同, 变为 ON。 如果条件重新变为正常, 保持 ON。 I/O 模块验证操作对远程 I/O 站点模块也进行。 	S (有错误)	M9002	
SM62	报警器检测	OFF : 不被检测 ON : 被检测	<ul style="list-style-type: none"> 即使只有一个报警器 F 变为 ON, 也变为 ON。 	S (指令执行)	M9009	○
SM80	CHK 检测	OFF : 不被检测 ON : 被检测	<ul style="list-style-type: none"> 如果有错误被 CHK 指令检测到, 变为 ON。 如果条件重新变为正常, 保持 ON。 	S (指令执行)	新增	

特殊继电器列表(续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM90	启动用于步转换 的看门狗定时器 (只有在 SFC 程序 存在时才被使能)	OFF :不启动 (看门狗定时器复位) ON :启动 (看门狗定时器启动)	对应于 SD90	U	M9108	○
SM91			对应于 SD91		M9109	
SM92			对应于 SD92		M9110	
SM93			对应于 SD93		M9111	
SM94			对应于 SD94		M9112	
SM95			对应于 SD95		M9113	
SM96			对应于 SD96		M9114	
SM97			对应于 SD97		新增	
SM98			对应于 SD98		新增	
SM99			对应于 SD99		新增	

(2) 系统信息

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM202	LEDOFF 命令	OFF → ON :LEDOFF	● 当此继电器从 OFF 变为 ON, 和 SD202 上每个位相对应的 LED 关闭。	U	新增	○
SM203	STOP 触点	STOP 状态	● 当处于 STOP 状态时, 变为 ON。	S (状态改变)	M9042	
SM204	PAUSE 触点	PAUSE 状态	● 当处于 PAUSE 状态时, 变为 ON。	S (状态改变)	M9041	
SM205	STEP-RUN 触点	STEP-RUN 状态	● 当处于 STEP-RUN 状态时, 变为 ON。	S (状态改变)	M9054	
SM206	PAUSE 使能线圈	OFF :PAUSE 无效 ON :PAUSE 有效	● 如果当远程 PAUSE 触点变为 ON 时此继电器变为 ON, 进入 PAUSE 状态。	U	M9040	远程
	设备元件测试请求接受状态	OFF :设备元件测试未被执行 ON :设备元件测试被执行	● 当设备元件测试模式在 GX Developer 上被执行时, 变为 ON。	S(请求)	新增	
SM210	时钟数据设置请求	OFF :忽略 ON :设置请求	● 当此继电器从 OFF 变为 ON 时, 在改变扫描周期的 END 指令执行结束后存储在从 SD210 到 SD213 的时钟数据被写入时钟设备元件。	U	M9025	○
SM211	时钟数据错误	OFF :无错误 ON :错误	● 当在时钟数据(SD210 到 SD213)中产生错误时变为 ON, 在没有错误被检测到时为 OFF。	S(请求)	M9026	
SM212	时钟数据显示	OFF :忽略 ON :显示	● 在 CPU 模块前面的 LED 显示器上显示时钟数、月、日、小时、分钟和秒。 (只对 Q3ACPU 和 Q4ACPU 有效)	U	M9027	Q3A Q4A Q4AR
SM213	时钟数据读请求	OFF :忽略 ON :读请求	● 当此继电器变为 ON 时, 时钟数据以 BCD 值的形式读到 SD210 和 SD213 中。	U	M9028	○+Rem
SM240	第 1 CPU 复位标志	OFF :第 1 CPU 复位取消 ON :第 1 CPU 重新设置	● 当第 1 CPU 复位被取消时, 变为 OFF。 ● 当第 1 CPU 被重新设置(包括 PLC 从基板中移掉的情况)时, 变为 ON。 其它 PLC 也被设置为复位状态。	S (状态改变)	新增	QCPU 功能 版本 B
SM241	第 2 CPU 复位标志	OFF :第 2 CPU 复位取消 ON :第 2 CPU 重新设置	● 当第 2 CPU 的复位被取消时, 变为 OFF。 ● 当第 2 CPU 被重新设置(包括 PLC 从基板中移掉的情况)时, 变为 ON。 其它 PLC 进入“MULTICPUDOWN”(错误代码:7000)。			
SM242	第 3 CPU 复位标志	OFF :第 3 CPU 复位取消 ON :第 3 CPU 重新设置	● 当第 3 CPU 的复位被取消时, 变为 OFF。 ● 当第 3 CPU 被重新设置(包括 PLC 从基板中移掉的情况), 变为 ON。 其它 PLC 进入“MULTICPUDOWN”(错误代码:7000)。			
SM243	第 4 CPU 复位标志	OFF :第 4 CPU 复位取消 ON :第 4 CPU 重新设置	● 当第 4 CPU 的复位被取消时, 变为 OFF。 ● 当第 4 CPU 被重新设置(包括 PLC 从基板中移掉的情况), 变为 ON。 其它 PLC 进入“MULTICPUDOWN”(错误代码:7000)。			
SM244	第 1 CPU 错误标志	OFF :第 1 CPU 正常 ON :第 1 CPU 处于停止错误	● 当第 1 CPU 为正常(包含连续错误), 变为 OFF。 ● 当第 1 CPU 处于停止错误, 变为 ON。			

特殊继电器列表(续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM245	第 2 CPU 错误标志	OFF :第 2 CPU 正常 ON :第 2 CPU 处于停止错误 错误	<ul style="list-style-type: none"> 当第 2 CPU 正常(包含连续错误)时, 变为 OFF。 当第 2 CPU 处于停止错误时, 变为 ON。 	S (状态改变)	新增	QCPU 功能 版本 B
SM246	第 3 CPU 错误标志	OFF :第 3 CPU 正常 ON :第 3 CPU 处于停止错误 错误	<ul style="list-style-type: none"> 当第 3 CPU 正常(包含连续错误)时, 变为 OFF。 当第 3 CPU 处于停止错误时, 变为 ON。 			
SM247	第 4 CPU 错误标志	OFF :第 4 CPU 正常 ON :第 4 CPU 处于停止错误 错误	<ul style="list-style-type: none"> 当第 4 CPU 正常(包含连续错误)时, 变为 OFF。 当第 4 CPU 处于停止错误时, 变为 ON。 			
SM250	读安装的最大 I/O 号码	OFF :忽略 ON :读	<ul style="list-style-type: none"> 当此继电器从 OFF 变为 ON 时, 将安装的 I/O 号码的最大值读到 SD250。 	U	新增	○+Rem
SM251	I/O 改变标志	OFF :无替换 ON :有替换	<ul style="list-style-type: none"> 在 SD251 中设置的被替换的 I/O 模块的头 I/O 号码被设置后, 当此继电器变为 ON 时, 在线 I/O 模块替换被使能。 (在每次设置时只有一块模块可以被替换。) 为替换一个处于 RUN 状态的 I/O 模块, 使用程序或外围设备元件来将此继电器变为 ON; 为替换一个处于 STOP 状态的 I/O 模块, 将继电器在一个外围设备元件的测试模式中变为 ON。 在 I/O 模块替换操作完成前不要在 RUN 和 STOP 状态之间切换。 	U (END)	M9094	Q2A (S1) Q3A Q4A Q4AR
SM252	I/O 改变 OK	OFF :替换被禁止 ON :替换被允许	<ul style="list-style-type: none"> 当 I/O 替换 OK 时, 变为 ON。 	S (END)	新增	
SM254	所有站点刷新命令	OFF :刷新到达的站点 ON :刷新所有站点	<ul style="list-style-type: none"> 使批量刷新有效(对低速周期也有效)。 用于指定是只接收到达的站点还是接收所有从站点。 	U (每个 END)	新增	QCPU
SM255	MELSECNET/10 模块 1 信息	OFF :活动网络 ON :备用网络	<ul style="list-style-type: none"> 为备用网络变为 ON。 (如果没有关于活动网络或备用网络的指定, 则假设为活动网络) 	S (初始化)	新增	○
SM256		OFF :读 ON :不读	<ul style="list-style-type: none"> 对从链接到 CPU 模块的刷新(B, W, 等), 表示是否从链接模块中读数据。 	U	新增	
SM257		OFF :写 ON :不写	<ul style="list-style-type: none"> 对从 CPU 模块到链接的刷新(B, W, 等), 表示是否向链接模块中写入数据。 	U	新增	
SM260	MELSECNET/10 模块 2 信息	OFF :活动网络 ON :备用网络	<ul style="list-style-type: none"> 为备用网络变为 ON。 (如果没有关于活动网络或备用网络的指定, 则假设为活动网络) 	S (初始化)	新增	
SM261		OFF :读 ON :不读	<ul style="list-style-type: none"> 对从链接到 CPU 模块的刷新(B, W, 等), 表示是否从链接模块中读数据。 	U	新增	
SM262		OFF :写 ON :不写	<ul style="list-style-type: none"> 对从 CPU 模块到链接的刷新(B, W, 等), 表示是否向链接模块中写入数据。 	U	新增	
SM265	MELSECNET/10 模块 3 信息	OFF :活动网络 ON :备用网络	<ul style="list-style-type: none"> 为备用网络变为 ON。 (如果没有关于活动网络或备用网络的指定, 则假设为活动网络)。 	S (初始化)	新增	
SM266		OFF :读 ON :不读	<ul style="list-style-type: none"> 对从链接到 CPU 模块的刷新(B, W, 等), 表示是否从链接模块中读数据。 	U	新增	
SM267		OFF :写 ON :不写	<ul style="list-style-type: none"> 对从 CPU 模块到链接的刷新(B, W, 等), 表示是否向链接模块中写入数据。 	U	新增	
SM270	MELSECNET/10 模块 4 信息	OFF :活动网络 ON :备用网络	<ul style="list-style-type: none"> 为备用网络变为 ON。 (如果没有关于活动网络或备用网络的指定, 则假设为活动网络) 	S (初始化)	新增	
SM271		OFF :读 ON :不读	<ul style="list-style-type: none"> 对从链接到 CPU 模块的刷新(B, W, 等), 表示是否从链接模块中读数据。 	U	新增	
SM272		OFF :写 ON :不写	<ul style="list-style-type: none"> 对从 CPU 模块到链接的刷新(B, W, 等), 表示是否向链接模块中写入数据。 	U	新增	
SM280	CC-Link 错误	OFF :正常 ON :错误	<ul style="list-style-type: none"> 当在安装的任一个 CC-Link 模块中检测到 CC-Link 错误时变为 ON, 当正常的操作被恢复时变为 OFF。 	S (状态改变)	新增	QCPU 远程
			<ul style="list-style-type: none"> 当在任一个安装的 CC-Link 模块中检测到 CC-Link 错误时变为 ON。 如果从那以后条件恢复为正常, 保持 ON。 	S (Error)	新增	QnA
SM320	存在/不存在 SFC 程序	OFF :SFC 程序不存在 ON :SFC 程序存在	<ul style="list-style-type: none"> 当有 SFC 程序产生时, 变为 ON。 当没有 SFC 程序产生时, 为 OFF。 	S (初始化)	M9100	○

特殊继电器列表 (续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM321	启动/停止 SFC 程序	OFF :SFC 程序停止 ON :SFC 程序启动	<ul style="list-style-type: none"> 初始值设置为和 SM320 的值相同。 (如果存在 SFC 程序则自动变为 ON。) 当此继电器从 OFF 变为 ON, 则启动 SFC 程序。 当此继电器从 ON 变为 OFF, 则停止 SFC 程序。 	S(初始化) U	M9101 格式改变	○
SM322	SFC 程序启动状态	OFF :初始启动 ON :重新启动	<ul style="list-style-type: none"> 在 PLC 参数对话框的 SFC 设置中的 SFC 程序启动模式被设置为初始值。 在初始启动时:OFF 在连续启动时:ON 	S (初始化) U	M9102 格式改变	
SM323	存在/不存在对整个块的连续转换	OFF :连续转换无效 ON :连续转换有效	<ul style="list-style-type: none"> 对 SFC 数据设备元件的“连续转换位”没有设置的块进行存在/不存在连续转换设置。 	U	M9103	
SM324	连续转换防止标志	OFF :当转换被执行时 ON :当没有转换时	<ul style="list-style-type: none"> 当处于连续转换模式操作中或处于连续转换时为 OFF, 当连续转换不被执行时变为 ON。 在无连续转换模式的操作中为常 ON。 	S (指令执行)	M9104	
SM325	块停止时的输出模式	OFF :OFF ON :保持	在块停止时选择是否保持活动步的线圈输出。 <ul style="list-style-type: none"> 作为初始值, 当线圈输出为 OFF 时参数中块停止时的输出模式为 OFF, 当线圈输出被保持时变为 ON。 当此继电器为 OFF 时所有线圈输出变为 OFF。 当此继电器变为 ON 时线圈输出被保持。 	S (初始化) U	M9196	
SM326	SFC 设备清除模式	OFF :清除设备 ON :保持设备	当存在 SFC 程序时, 在顺序程序或 SFC 程序被修改后, 停止的 CPU 重新运行时, 选择设备的状态。	U	新增	
SM327	结束步执行时的输出	OFF :OFF ON :保持	在从 STOP 到程序写到 RUN 的切换时刻, 选择设备状态(除步进继电器外的所有设备)。	S (初始化) U	新增	
SM330	用于低速执行类型程序的操作模式	OFF :异步模式 ON :同步模式	选择低速执行类型程序是按照异步模式还是同步模式执行。 <ul style="list-style-type: none"> 异步模式(此继电器被变为 OFF。) 在额外时间内, 低速执行类型程序的操作被连续执行的模式。 同步模式(此继电器被切换变为 ON。) 如果存在额外的时间, 低速执行类型程序的操作不被连续执行的模式, 以及从下一个扫描周期开始被执行的执行的操作 	U(END)	新增	
SM331	正常 SFC 程序执行状态	OFF :没有被执行 ON :正被执行	<ul style="list-style-type: none"> 表示正常的 SFC 程序是否正被执行。 被用作 SFC 控制指令执行互锁。 	S (状态改变)	新增	QCPU 序列号 04122 或 更晚
SM332	程序执行管理 SFC 程序执行状态	OFF :没有被执行 ON :正被执行	<ul style="list-style-type: none"> 表示程序执行管理 SFC 程序是否正被执行。 被用作 SFC 控制指令执行互锁。 			
SM390	访问执行标志	ON 表示对智能功能模块访问的完成	<ul style="list-style-type: none"> 存储前智能功能模块访问指令被执行的状态。 (在智能功能模块访问指令被再次执行时覆盖此数据。) 被用户在程序中用作完成位。 	S (状态改变)	新增	QCPU

特殊继电器列表

(3) 系统时钟/计数器

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM400	常 ON	ON 	● 正常为 ON。	S (每个 END 处理)	M9036	○
SM401	常 OFF	ON 	● 正常为 OFF。	S (每个 END 处理)	M9037	
SM402	只在 RUN 后 ON 一个扫描周期	ON 	● 在 RUN 后, ON 一个扫描周期。 ● 此连接只能被用于扫描执行类型程序。 	S (每个 END 处理)	M9038	
SM403	在 RUN 后, 只 OFF 一个扫描周期	ON 	● 在 RUN 后, 只 OFF 一个扫描周期。 ● 此连接只能被用于扫描执行类型程序。 	S (每个 END 处理)	M9039	
SM404	低速执行类型程序在 RUN 后, ON 一个扫描周期	ON 	● 在 RUN 后, ON 一个扫描周期。 ● 此连接只能用于低速执行类型程序。	S (每个 END 处理)	新增	
SM405	低速执行类型程序在 RUN 后, 只 OFF 一个扫描周期	ON 	● 在 RUN 后, 只 OFF 一个扫描周期。 ● 此连接只能被用于低速执行类型程序。	S (每个 END 处理)	新增	
SM409	0.01 秒时钟		● 以 5ms 为间隔在 ON 和 OFF 之间循环改变。 ● 当 PLC 供电电源变为 OFF 或 CPU 模块被复位时, 从 OFF 开始启动。 ● 注意, 在程序的执行过程中当指定的时间用完时 ON-OFF 状态改变。	S (状态改变)	新增	QCPU
SM410	0.1 秒时钟		● 在每个指定的时间间隔重复的在 ON 和 OFF 之间改变。 ● 当 PLC 供电电源变为 OFF 或有 CPU 模块被复位时, 从 OFF 开始启动。 ● 注意, 在程序的执行过程中当指定的时间用完时 ON-OFF 状态改变。	S (状态改变)	M9030	○
SM411	0.2 秒时钟				M9031	
SM412	1 秒时钟				M9032	
SM413	2 秒时钟				M9033	
SM414	2n 秒时钟		● 此继电器按照在 SD414 中指定的时间 (单位: S) 间隔在 ON 和 OFF 之间交替改变。 ● 当 PLC 供电电源变为 OFF 或有 CPU 模块被复位时, 从 OFF 开始启动。 ● 注意, 在程序的执行过程中当指定的时间用完时 ON-OFF 状态改变。	S (状态改变)	M9034 格式改变	○
SM415	2n (ms) 时钟		● 此继电器按照在 SD415 中指定的时间 (单位: ms) 间隔在 ON 和 OFF 之间交替改变。 ● 当 PLC 供电电源变为 OFF 或有 CPU 模块被复位时, 从 OFF 开始启动。 ● 注意, 在程序的执行过程中当指定的时间用完时 ON-OFF 状态改变。	S (状态改变)	新增	QCPU

特殊继电器列表 (续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU	
SM420	0 号用户定时时钟		SM421	1 号用户定时时钟	S (每个 END 处理)	M9020	○
SM422	2 号用户定时时钟		M9021				
SM423	3 号用户定时时钟		M9022				
SM424	4 号用户定时时钟		M9023				
SM430	5 号用户定时时钟		M9024				
SM431	6 号用户定时时钟			S (每个 END 处理)	新增		
SM432	7 号用户定时时钟						
SM433	8 号用户定时时钟						
SM434	9 号用户定时时钟						

(4) 扫描信息

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM510	低速程序执行标志	OFF : 已完成或未被执行 ON : 执行进行中	● 变为 ON, 当低速执行类型程序被执行时	S (每个 END 处理)	新增	○
SM551	读取模块服务间隔	OFF : 忽略 ON : 读	● 当此继电器从 OFF 变为 ON, 由 SD550 的模块服务间隔被读到从 SD551 到 SD552。	U	新增	○+Rem

(5) 存储卡

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM600	存储卡 (A) 可用标志	OFF : 不可用 ON : 可用	● 存储卡 (A) 准备好由用户使用时为 ON。	S (初始化)	新增	○
SM601	存储卡 (A) 保护标志	OFF : 无保护 ON : 有保护	● 当存储卡 (A) 保护切换变为 ON 时, 变为 ON。	S (初始化)	新增	
SM602	驱动器 1 标志	OFF : 无驱动器 1 ON : 驱动器 1 存在	● 当安装的存储卡 (A) 为 RAM 时, 变为 ON。	S (初始化)	新增	
SM603	驱动器 2 标志	OFF : 无驱动器 2 ON : 驱动器 2 存在	● 当安装的存储卡 (A) 为 ROM 时, 变为 ON。	S (初始化)	新增	
SM604	存储卡 (A) 正被使用标志	OFF : 没有被使用 ON : 处于使用中	● 当存储卡 (A) 被使用时, 变为 ON。	S (状态改变)	新增	
SM605	存储卡 (A) 移掉/插入禁止标志	OFF : 移掉/插入允许 ON : 移掉/插入禁止	● 当存储卡 (A) 不能被插入或移掉时, 变为 ON。	U	新增	
SM609	存储卡移掉/插入允许标志	OFF : 移掉/插入禁止 ON : 移掉/插入允许	● 由用户变为 ON 以允许移掉/插入存储卡。 ● 在存储卡被移掉后由系统变为 OFF。 ● 此触点只有在 SM604 和 SM605 为 OFF 时才能使用。	U/S	新增	QCPU
SM620	存储卡 B 可用标志	OFF : 不可用 ON : 可用	● 常 ON ● 当存储卡 B 准备好由用户使用时为 ON。	S (初始化)	新增	Q2A (S1) Q3A Q4A Q4AR
SM621	驱动器 3/4 保护标志	OFF : 无保护 ON : 有保护	● 常 OFF	S (初始化)	新增	QCPU
	存储卡 B 保护标志	OFF : 无保护 ON : 有保护	● 当存储卡 B 保护开关变为 ON 时, 变为 ON	S (初始化)	新增	Q2A (S1) Q3A Q4A Q4AR

特殊继电器列表(续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM622	驱动器 3 标志	OFF : 无驱动器 3 ON : 驱动器 3 存在	● 常 ON	S(初始化)	新增	QCPU
			● 当驱动器 3(卡 2RAM 区域) 存在时, 变为 ON。	S(初始化)	新增	Q2A(S1) Q3A Q4A Q4AR
SM623	驱动器 4 标志	OFF : 无驱动器 4 ON : 驱动器 4 存在	● 常 ON	S(初始化)	新增	QCPU
			● 当驱动器 4(卡 2ROM 区域) 存在时, 变为 ON。	S(初始化)	新增	Q2A(S1)
SM624	存储卡 B 使用中标志	OFF : 未被使用 ON : 使用中	● 当存储卡 B 使用中时, 变为 ON。	S (状态改变)	新增	Q3A Q4A Q4AR
SM625	存储卡 B 移掉/插入禁止标志	OFF : 允许移掉/插入 ON : 禁止移掉/插入	● 当存储卡 B 不能被插入或移掉时, 变为 ON。	U	新增	
SM640	文件寄存器使用	OFF : 文件寄存器未被使用 ON : 文件寄存器使用中	● 当文件寄存器使用中时, 变为 ON。	S (状态改变)	新增	○
SM650	注释使用	OFF : 文件寄存器未被使用 ON : 文件寄存器使用中	● 当注释文件使用中时, 变为 ON。	S (状态改变)	新增	
SM660	引导操作	OFF : 内部存储器执行 ON : 引导操作处理中	● 当引导操作处理中时, 变为 ON。 ● 如果引导指定变为 OFF 时, 变为 OFF。	S (状态改变)	新增	
SM672	存储卡 A 文件寄存器访问范围标志	OFF : 处于访问范围内 ON : 处于访问范围外	● 当访问处于存储卡 A 的文件寄存器 R 的访问范围区域外时, 变为 ON。 (在 END 处理中设置) ● 在用户程序中复位	S/U	新增	
SM673	存储卡 B 文件寄存器访问范围标志	OFF : 处于访问范围内 ON : 处于访问范围外	● 当访问处于存储卡 B 的文件寄存器 R 的访问范围区域外时, 变为 ON。 (在 END 处理中设置) ● 在用户程序中复位	S/U	新增	Q2A(S1) Q3A Q4A Q4AR

(6) 和特殊继电器相关的指令

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM700	进位标志	OFF :进位 OFF ON :进位 ON	● 用于应用指令的进位标志	S (指令执行)	M9012	○
SM701	输出字符数选择	OFF :16 字符输出 ON :输出直到 NUL	● 当 SM701 为 OFF, 输出 16 字符的 ASCII 码。 ● 当 SM701 变为 ON, 执行输出, 直到碰到 ASCII 码的 NUL (00H) 码。	U	M9049	
SM702	搜索方法	OFF :搜索下一个 ON :2-部分搜索	● 指定搜索指令使用的方法。 ● 数据必须按照 2 部搜索的要求排列。	U	新增	
SM703	排序	OFF :升序 ON :降序	● 排序指令用于指定数据是按照升序还是按照降序排列。	U	新增	
SM704	块比较	OFF :没有找到匹配 ON :所有的都匹配	● 当 BKMP 指令的所有数据条件都满足时, 变为 ON	S (指令执行)	新增	
SM707	实数指令处理类型的选择	OFF :基于速度 ON :基于正确度	● 当 SM707 为 OFF, 实数指令高速执行。 ● 当 SM707 变为 ON, 实数指令按高正确度执行。	U	新增	Q4AR
SM710	CHK 指令优先级标志	OFF :条件优先 ON :模式优先	● 当为 OFF 时保持初始设置。 ● 当变为 ON 时进行 CHK 优先级更新。	S (指令执行)	新增	○
SM711	分开传送状态	OFF :不是处于分开处理 ON :处于分开处理	● 在 AD57 (S1) 的处理中, 当屏幕裂开用于传送时变为 ON, 当裂开处理结束时变为 OFF。	S (指令执行)	M9065	QnA
SM712	传送处理选择	OFF :批处理 ON :分开处理	● 在 AD57 (S1) 的处理中, 当防水屏幕被分开用于传送时变为 ON。	S (指令执行)	M9066	
SM714	通信请求注册区域 BUSY 信号	OFF :使能对远程终端模块的通信请求 ON :禁止对远程终端模块的通信请求	● 用于决定对连接到 AJ71PT32-S3 的远程终端模块的通信请求是否被执行。	S (指令执行)	M9081	
SM715	EI 标志	OFF :处于 DI ON :处于 EI	● 当 EI 指令被执行时变为 ON。	S (指令执行)	新增	○
SM720	注释读完成标志	OFF :注释读没有被完成 ON :注释读完成	● 当 COMRD 或 PRC 指令的处理完成时, 变为 ON 一个扫描周期。	S (状态改变)	新增	QCPU
SM721	文件正被访问	OFF :文件没有被访问 ON :文件正被访问	● 切换变为 ON, 当一个文件正被 S, FWRITE, S, FREAD, COMRD, PRC, 或 LEDC 指令访问时。	S (状态改变)	新增	
SM722	BIN/DBIN 指令错误禁止标志	OFF :执行错误检测 ON :不执行错误检测	● 当 “OPERATIONERROR” 由于 BIN 或 DBIN 指令而被禁止时, 变为 ON	U	新增	
SM730	用于 CC-Link 通信请求注册区域的 BUSY 信号	OFF :使能和智能设备站点的通信请求 ON :禁止和智能设备站点的通信请求	● 用于决定是否使能和连接到 CC-Link 模块的智能设备站点的通信请求。	S (指令执行)	新增	QnA
SM736	PKEY 指令执行处于处理中的标志	OFF :指令未被指令 ON :指令执行	● 当 PKEY 指令正被执行时, ON。 当 CR 为输入时或者当输入字符串达到 32 个字符时, 变为 OFF。	S (指令执行)	新增	○
SM737	PKEY 指令的键盘输入接收标志	OFF :使能键盘输入接收 ON :禁止键盘输入接收	● 当执行键盘输入时, 变为 ON。 当键盘输入已经被存储到 CPU 时, 变为 OFF。	S (指令执行)	新增	

特殊继电器列表 (续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM738	MSG 指令接收标志	OFF :指令未被执行 ON :指令被执行	● 当 MSG 指令被执行时, 变为 ON。	S (指令执行)	新增	○
SM774	PID 无扰处理 (用于完整微分)	OFF :强制匹配 ON :不强制匹配	● 指定在手动模式中设置值 (SV) 是否和过程值 (PV) 相匹配。	U	新增	
SM775	在 COM 指令执行过程中链接刷新处理的选择	OFF :执行链接刷新 ON :无链接刷新被执行	● 选择是只执行用于 COM 指令执行的一般数据处理还是将链接刷新处理一并执行。	U	新增	
		OFF :执行所有的刷新处理 ON :执行设置在 SD778 上的刷新	● 当 COM 指令被执行时选择是所有的刷新处理都被执行还是只执行设置在 SD778 上的刷新处理。	U	新增	QCPU 序列号 04012 或更晚
SM776	在 CALL 上使能/禁止本地设备	OFF :本地设备禁止 ON :本地设备使能	● 设置在 CALL 指令执行时被调用的子程序的本地设备是否有效。	U (状态改变)	新增	○
SM777	在中断程序中使能/禁止本地设备	OFF :本地设备禁止 ON :本地设备使能	● 设置在中断程序执行中本地设备是否有效。	U (状态改变)	新增	
SM780	CC-Link 专用的可执行指令	OFF :CC-Link 专用的可执行指令 ON :CC-Link 专用的不可执行指令	● 当能被同时执行的 CC-Link 专用指令数达到 32 时变为 ON。当指令数小于 32 时变为 OFF。	U (状态改变)	新增	QnA
SM794	PID 无扰处理 (用于完整的微分)	OFF :强制匹配 ON :不强制匹配	● 指定在手动模式中是否将设置值 (SV) 和过程值 (PV) 匹配。	U	新增	QCPU 序列号 05032 或更晚

(7) 调试

号码	名称	含义	解释	设置者 (当被设置时)	设置者 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM800	跟踪准备	OFF :没有准备好 ON :已准备好	● 当跟踪准备完成时变为 ON。	S (状态改变)	新增	QCPU
	采样跟踪准备		● 当采样跟踪已准备好时变为 ON。	S (状态改变)	新增	QnA
SM801	跟踪开始	OFF :挂起 ON :开始	● 当此继电器切换变为 ON 时跟踪开始。 ● 当此继电器变为 OFF 时跟踪挂起。 (所有相关的特殊 Ms 变为 OFF。)	U	M9047	QCPU
	采样跟踪开始		● 当此变为 ON 时采样跟踪开始 ● 当为 OFF 时挂起(相关特殊 M 全为 OFF)	U	M9047	QnA
SM802	跟踪执行进行中	OFF :挂起 ON :开始	● 在跟踪执行过程中切换变为 ON	S (状态改变)	M9046	QCPU
	采样跟踪执行在进行中		● 在采样跟踪执行过程中变为 ON。	S (状态改变)	M9046	QnA
SM803	跟踪触发器	OFF → ON :开始	● 当此继电器从 OFF 切换变为 ON 时跟踪被触发。(和跟踪指令执行状态相同)	U	M9044	QCPU
	采样跟踪触发器		● 当此触发器从 OFF 变为 ON 时, 采样跟踪触发器变为 ON。(和 STRA 指令执行状态相同)	U	M9044	QnA
SM804	跟踪触发器之后	OFF :不在触发器之后 ON :在触发器之后	● 在跟踪被触发之后切换变为 ON。	S (状态改变)	新增	QCPU
	采样跟踪触发器之后		● 在采样跟踪触发后变为 ON。	S (状态改变)	新增	QnA
SM805	跟踪完成	OFF :未完成 ON :结束	● 在跟踪完成时切换变为 ON。	S (状态改变)	9043	QCPU
	采样跟踪完成		● 在采样跟踪完成时变为 ON。	S (状态改变)	9043	QnA

特殊继电器列表 (续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM806	状态闭锁准备	OFF : 没有准备好 ON : 已准备好	● 当状态闭锁已准备好时, 变为 ON。	S (状态改变)	新增	QnA
SM807	状态闭锁命令	OFF → ON : 闭锁	● 运行状态闭锁命令。	U	新增	
SM808	状态闭锁完成	OFF : 闭锁没有完成 ON : 闭锁完成	● 当状态闭锁完成时, 变为 ON。	S (状态改变)	9055	
SM809	状态闭锁清除	OFF → ON : 清除	● 使能下一个状态闭锁。	U	新增	
SM810	程序跟踪准备	OFF : 没有准备好 ON : 已准备好	● 当程序跟踪已准备好时, 变为 ON。	S (状态改变)	新增	QnA
SM811	启动程序跟踪	OFF : 挂起 ON : 启动	● 当其变为 ON 时, 程序跟踪启动。 ● 当 OFF 时被挂起(相关的特殊 M 全为 OFF)	S (状态改变)	新增	
SM812	程序跟踪执行进行中	OFF : 挂起 ON : 启动	● 在程序跟踪执行进行中为 ON。	U	新增	
SM813	程序跟踪触发器	OFF → ON : 启动	● 当其从 OFF 变为 ON 时, 程序跟踪触发器变为 ON。(和 PTR 指令执行状态一样)	S (状态改变)	新增	
SM814	在程序跟踪触发器之后	OFF : 不在触发器之后 ON : 在触发器之后	● 在程序跟踪触发器之后变为 ON。	S (状态改变)	新增	
SM815	程序跟踪完成	OFF : 没有完成 ON : 结束	● 在程序跟踪结束时变为 ON。	S (状态改变)	新增	○
SM820	步跟踪准备	OFF : 没有准备好 ON : 已准备好	● 在程序跟踪注册后准备好时, 变为 ON。	U	新增	
SM821	步跟踪启动	OFF : 挂起 ON : 启动	● 当此变为 ON 时, 步跟踪启动。 ● 当 OFF 时挂起(相关特殊 M 全为 OFF)。	S (状态改变)	M9182 格式改变	
SM822	步跟踪执行进行中	OFF : 挂起 ON : 启动	● 当步跟踪执行处于进行中时, 变为 ON。 ● 在结束或挂起时变为 OFF。	S (状态改变)	M9181	
SM823	在步跟踪触发器之后	OFF : 不在触发器之后 ON : 在第一个触发器之后	● 即使在执行的步跟踪只有一个块被触发, 也变为 ON。 ● 在步跟踪开始后, 变为 OFF。	S (状态改变)	新增	
SM824	在步跟踪触发器之后	OFF : 不在所有的触发器之后 ON : 在所有的触发器之后	● 如果被执行的步跟踪中所有的块都被触发, 则变为 ON。 ● 当步跟踪开始后, 变为 OFF。	S (状态改变)	新增	QnA
SM825	步跟踪完成	OFF : 没有完成 ON : 结束	● 在步跟踪完成时变为 ON。 ● 当步跟踪开始后变为 OFF。	S (状态改变)	M9180	
SM826	跟踪错误	OFF : 正常 ON : 错误	● 如果在跟踪执行过程中发生错误切换变为 ON。	S (状态改变)	新增	
	采样跟踪错误		● 如果在采样跟踪执行过程中发生错误, 变为 ON。	S (状态改变)	新增	
SM827	状态闭锁错误	OFF : 正常 ON : 错误	● 如果在状态闭锁执行过程中发生错误, 变为 ON。	S (状态改变)	新增	
SM828	程序跟踪错误	OFF : 正常 ON : 错误	● 如果在程序跟踪执行过程中发生错误, 变为 ON。	S (状态改变)	新增	

(8) 闭锁区域

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM900	电源切断文件	OFF : 无电源切断文件 ON : 有电源切断文件	● 如果在电源中断时有文件正处于被访问时, 变为 ON。	S/U (状态改变)	新增	QnA
SM910	RKEY 注册标志	OFF : 键盘输入没有被注册 ON : 键盘输入被注册	● 在键盘输入被注册时变为 ON。 如果键盘输入没有被注册则为 OFF。	S (指令执行)	新增	

(9) A 到 Q/QnA 转换的对应

特殊继电器 SM1000 到 SM1255 是在 A 到 Q/QnA 转换后和 ACPU 的特殊继电器 M9000 到 M9255 对应的继电器。

所有这些继电器都由系统控制，这样用户就不能在程序中改变它们的 ON/OFF。

如果用户希望改变这些继电器的 ON/OFF，那么就要修改程序以使用 QCPU/QnACPU 特殊继电器。

可是，对于 SM1084 和 SM1200 到 SM1255，如果用户可以在转换前将某些特殊继电器 M9084 和 M9200 到 M9255 变为 ON/OFF，那么用户也能在转换后将 SM1084 和 SM1200 到 SM1255 中的相应继电器变为 ON/OFF。

对于 ACPU 中的特殊继电器的详细信息，参见每个 CPU 的用户手册，和 MELSECNET 或 MELSECNET/B 数据链接系统参考手册。

要点

当修改后的特殊继电器被用于 QCPU 时，处理时间可能会长一些。当不使用修改后的特殊继电器时，GX Developer 参数中的 PC 系统设置中的“A-系列 CPU 兼容性设置”不被检查。

备注

以下是对修改栏中关于特殊继电器的新增加的解释。

- ① 当一个特殊继电器被用于修改时，软元件号应改为所提供的 QCPU/QnACPU 特殊继电器。
- ② 出现 ☐ 时，修改后的特殊继电器可以被用于此软元件号。
- ③ 出现 ☒ 时，此软元件号不能用于 QCPU/QnACPU。

特殊继电器列表

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义	详细信息	可用的 CPU
M9000	SM1000	—	保险丝烧坏	OFF : 正常 ON : 模块有保险丝被烧坏	<ul style="list-style-type: none">● 当有一个或多个输出模块的保险丝被烧坏时，变为 ON。● 如果此后条件恢复为正常，保持为 ON。● 远程 I/O 站的输出模块也进行保险丝状况检查。	○
M9002	SM1002	—	I/O 模块确认错误	OFF : 正常 ON : 有错误	<ul style="list-style-type: none">● 如果 I/O 模块的状态和系统上电时的状态不同，变为 ON。● 如果此后条件恢复为正常，保持 ON。● 对远程 I/O 站的模块也进行 I/O 模块确认。● 复位只在特殊寄存器 SD1116 到 SD1123 被复位时才被允许。	
M9004	SM1004	—	NIMI 链接错误	OFF : 正常 ON : 有错误	<ul style="list-style-type: none">● 如果即使只有一个被安装的 AJ71PT32 (S3) 模块被发现有 MINI (S3) 链接错误，也变为 ON。● 如果此后条件恢复为正常，保持 ON。	QnA
M9005	SM1005	—	AC DOWN 检测	OFF : AC DOWN 没有被检测 ON : AC DOWN 被检测	<ul style="list-style-type: none">● 在使用 AC 电源供电模块时，发生 20ms 内的瞬时电源故障时，变为 ON。● 当电源切换为 OFF 时复位，然后变为 ON。	○
					<ul style="list-style-type: none">● 变为 ON，在使用 DC 电源供电模块时，发生 10ms 内的瞬时电源故障。● 当电源切换为 OFF 时复位，然后变为 ON。● 在使用 DC 电源供电模块时，发生 1ms 内的瞬时电源故障时，变为 ON。● 当电源切换为 OFF 时复位，然后变为 ON。	QnA

特殊继电器列表 (续表)

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义	详细信息	可用的 CPU
M9006	SM1006	—	电池电压低	OFF : 正常 ON : 电池电压低	● 当电池电压下降或低于指定值时, 变为 ON。 ● 如果电池电压从那以后恢复到正常时, 则变为 OFF。	○
M9007	SM1007	—	电池电压低闭锁	OFF : 正常 ON : 电池电压低	● 当电池电压下降或低于指定值时, 变为 ON。 ● 如果电池电压从那以后恢复到正常, 保持 ON。	
M9008	SM1008	SM1	自诊断错误	OFF : 电池电压低 ON : 有错误	● 当有错误被当作自诊断结果被发现时, 变为 ON。	
M9009	SM1009	SM62	报警器检测	OFF : 无 F 号码被检测到 ON : F 号码被检测到	● 当 OUT F 指令的 SET F 被执行时, 变为 ON。 ● 当 SD1124 数据被清零时, 切换为 OFF。	
M9011	SM1011	SM56	操作错误标志	OFF : 无错误 ON : 有错误	● 当在应用指令的执行过程中发生操作错误时, 变为 ON。 ● 如果从那以后条件恢复到正常, 则保持为 ON。	
M9012	SM1012	SM700	进位标志	OFF : 进位 OFF ON : 进位 ON	● 进位标志用在应用指令中。	
M9016	SM1016		数据存储区清除标志	OFF : 忽略 ON : 输出清除	● 当 SM1016 为 ON 时, 从计算机上清除包括处于远程运行模式的闭锁区域 (除特殊继电器和特殊寄存器) 内的数据存储器。	○
M9017	SM1017		数据存储区清除标志	OFF : 忽略 ON : 输出清除	● 当 SM1016 为 ON 时, 从计算机上清除包括处于远程运行模式的没有被闭锁的区域 (除特殊继电器和特殊寄存器) 内的数据存储器。	
M9020	SM1020	—	0 号用户定时时钟		● 按照预先设定的扫描间隔重复 ON/OFF 的继电器。 ● 当电源变为 ON 或执行复位操作时, 时钟从 OFF 开始启动。 ● 由 DUTY 指令设定 ON/OFF 的间隔。 	
M9021	SM1021	—	1 号用户定时时钟			
M9022	SM1022	—	2 号用户定时时钟			
M9023	SM1023	—	3 号用户定时时钟			
M9024	SM1024	—	4 号用户定时时钟			
M9025	SM1025	—	时钟数据设置请求	OFF : 忽略 ON : 设置请求当前被使用	● 在 SM1025 从 OFF 变为 ON 的扫描周期内, 在 END 指令被执行后, 将从 SD1025 到 SD1028 的时钟数据写到时钟元件内。	Q3A Q4A Q4AR
M9026	SM1026	—	时钟数据错误	OFF : 无错误 ON : 有错误	● 被时钟数据 (SD1025 到 SD1028) 错误切换为 ON。	
M9027	SM1027	—	时钟数据显示	OFF : 忽略 ON : 显示	● 时钟数据被从 SD1025 到 SD1028 内读出, 并且月, 日, 小时, 分钟和秒被显示在 CPU 模块前面的 LED 显示器上。	
M9028	SM1028	—	时钟数据读请求	OFF : 忽略 ON : 读请求	● 当 SD1028 为 ON 时, 以 BCD 码的形式将时钟数据读入 SD1025 到 SD1028 内。	○
M9029	SM1029		数据通信请求的批处理	OFF : 批处理没有被执行 ON : 批处理被执行	● 在此扫描周期的 END 处理内, SM1029 继电器变为 ON, 使用顺序程序来处理在此扫描周期内接收到的所有数据通信请求。 ● 在运行过程中, 数据通信请求的批处理可以被切换为 ON 和 OFF。 ● 默认为 OFF (按照数据通信请求被接收到的次序, 每个 END 处理一次处理一个通信请求)。	
M9030	SM1030	—	0.1 秒时钟		● 0.1 秒, 0.2 秒, 1 秒和 2 秒时钟被创建。 ● 并不是每个扫描周期都切换为 ON 或 OFF, 但是, 如果相应的时间已用完, 即使是在扫描周期内也会切换为 ON 和 OFF。 ● 从 OFF 开始启动, 当 PLC 电源变为 ON 或 CPU 模块被执行复位。	
M9031	SM1031	—	0.2 秒时钟			
M9032	SM1032	—	1 秒时钟			
M9033	SM1033	—	2 秒时钟			

*: 1 分钟时钟表示 ACPU 的特殊继电器 (M9034) 的名称。

特殊继电器列表 (续表)

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义	详细信息	可用的 CPU
M9034	SM1034	—	2n 分钟时钟 (1 分钟时钟) *		<ul style="list-style-type: none">根据在 SD414 中设定的秒数 (默认:n=30) 在 ON 和 OFF 之间交替切换。并不是在每个扫描周期变为 ON 或 OFF, 但是, 即使是在扫描周期中, 如果相应的时间已经用完, 也会变为 ON 或 OFF。当 PLC 供电电源变为 ON 或 CPU 模块执行复位操作时, 从 OFF 启动。	○
M9036	SM1036	—	常 ON	ON OFF	<ul style="list-style-type: none">在顺序程序中用作初始化和应用指令的空触点。	
M9037	SM1037	—	常 OFF	ON OFF	<ul style="list-style-type: none">SM1038 和 SM1037 切换为 ON 和 OFF, 而不管 CPU 模块前部钥匙开关的位置。SM1038 和 SM1039 处于和钥匙开关处于 STOP 位置以外的 RUN 状态一样的条件, 并切换为 OFF 和 ON。切换到 OFF, 如果钥匙开关位于 STOP 位置。如果钥匙开关不是位于 STOP 位置, 则 SM1038 只 ON 一个扫描周期, SM1039 只 OFF 一个扫描周期。	
M9038	SM1038	—	在 RUN 后, 只 ON 一个扫描周期	ON OFF		
M9039	SM1039	—	RUN 标志 (在 RUN 后, 只 OFF 一个扫描周期)	ON OFF		
M9040	SM1040	SM206	PAUSE 许可线圈	OFF : PAUSE 不允许 ON : PAUSE 许可	<ul style="list-style-type: none">当 RUN 钥匙开关位于 PAUSE 位置或远程 PAUSE 触点已经变为 ON, 并且如果 SM204 为 ON, PAUSE 模式被设定并且 SM206 被变为 ON。	
M9041	SM1041	SM204	USE 状态触点	OFF : PAUSE 无效 ON : PAUSE 有效		
M9042	SM1042	SM203	STOP 状态触点	OFF : STOP 无效 ON : STOP 有效	<ul style="list-style-type: none">切换为 ON, 当 RUN 钥匙开关或 RUN/STOP 开关位于 STOP 位置。	
M9043	SM1043	SM805	采样跟踪完成	OFF : 采样跟踪处理中 ON : 采样跟踪完成	<ul style="list-style-type: none">在 [STRA] 指令执行后, 当采样跟踪执行完由参数预先指定的次数时, 切换为 ON。当 [STRAR] 指令被执行时复位。	
M9044	SM1044	SM803	采样跟踪	OFF → ON [STRA] 和执行一样 ON → OFF [STRAR] 和执行一样	<ul style="list-style-type: none">切换成 ON/OFF, SM803 能够执行 [STRA] / [STRAR] 指令。 (SM803 被外部设备强制切换为 ON/OFF) 当从 OFF 切换成 ON 时: [STRA] 指令 当从 ON 切换成 OFF 时: [STRAR] 指令 存储在 SD1044 中的值被用作采样跟踪的条件。 在扫描时, 在时间 → 时间 (10ms 单位)	
M9045	SM1045		看门狗定时器 (WDT) 复位	OFF : 不复位 WDT ON : 复位 WDT	<ul style="list-style-type: none">当 ZCOM 指令和数据通信请求批处理 (在扫描周期超过 200ms 时被使用) 被执行时, SM1015 继电器被切换为 ON 以复位 WDT。	
M9046	SM1046	SM802	采样跟踪	OFF : 跟踪未被执行 ON : 跟踪进行中	<ul style="list-style-type: none">在采样跟踪过程中被切换为 ON。	
M9047	SM1047	SM801	采样跟踪准备	OFF : 采样跟踪挂起 ON : 采样跟踪启动	<ul style="list-style-type: none">采样跟踪未被执行, 除非 SM801 变为 ON。当 SM801 变为 OFF 时, 采样跟踪被挂起。	
M9049	SM1049	SM701	输出字符数的选择	OFF : 输出, 直到碰到 NULL 代码 ON : 16 字符输出	<ul style="list-style-type: none">当 SM701 为 OFF 时, 输出字符直到 NULL (00H) 代码。当 SM701 为 ON 时, 输出 16 位字符的 ASCII 码。	
M9051	SM1051		CHG 指令执行禁止	OFF : 许可 ON : 无效	<ul style="list-style-type: none">切换为 ON 时 CHG 指令无效。当程序传送被请求时切换为 ON。当传送结束时自动切换为 OFF。	
M9052	SM1052		SEG 指令切换	OFF : 7SEG 部分显示 ON : I/O 局部刷新	<ul style="list-style-type: none">当 SM1052 为 ON 时, SEG 指令被作为 I/O 局部刷新指令。当 SM1052 为 OFF, SEG 指令被作为 7-SEG 显示指令。	
M9054	SM1054	SM205	STEP RUN 标志	OFF : STEP RUN 无效 ON : STEP RUN 有效	<ul style="list-style-type: none">当 RUN 钥匙开关位于 STEP RUN 位置时, 切换为 ON。	QnA
M9055	SM1055	SM808	状态闭锁完成标志	OFF : 未完成 ON : 完成	<ul style="list-style-type: none">当状态闭锁完成时变为 ON。由复位指令切换为 OFF。	
M9056	SM1056		主程序 P、I 设定请求	OFF : 除 P、I 设定被执行外的情况 ON : P、I 设定被执行	<ul style="list-style-type: none">在运行过程中, 其它程序 (例如, 主程序运行时的子程序) 的转换完成后提供 P、I 设定请求。当 P、I 设置完成后自动切换为 OFF。	○
M9057	SM1057		子程序 P、I 设定请求	OFF : 除 P、I 设定被执行外的情况 ON : P、I 设定被执行		

特殊继电器列表 (续表)

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义	详细信息	可用的 CPU
M9058	SM1058		主程序 P、I 设定完成	在 P、I 设定完成时立刻变为 ON	<ul style="list-style-type: none"> 在 P、I 设定完成后变为 ON，然后重新变为 OFF。 	○
M9059	SM1059		子程序 P、I 设定完成	在 P、I 设定完成时立刻变为 ON		
M9060	SM1060		子程序 2 P、I 设置请求	OFF : P、I 设定正被请求外的其它情况 ON : P、I 设定正被请求		
M9061	SM1061		子程序 3 P、I 设定请求	OFF : P、I 设定正被请求外的其它情况 ON : P、I 设定正被请求	<ul style="list-style-type: none"> 在运行过程中，其它程序(例如，主程序运行时的子程序)的转换完成后提供 P、I 设定请求。当 P、I 设置完成后自动切换为 OFF。 	QnA
M9065	SM1065	SM711	分开处理执行检测	OFF : 分开处理没有进行 ON : 处于分开处理中	<ul style="list-style-type: none"> 当防水屏幕被分开处理转移到 AD57 (S1)/AD58 时变为 ON，并在分开处理完成时变为 OFF。 	
M9066	SM1066	SM712	分开处理请求标志	OFF : 批处理 ON : 分开处理	<ul style="list-style-type: none"> 当防水屏幕被分开处理转移到 AD57 (S1)/AD58 时变为 ON。 	
M9070	SM1070		A8UPU/A8PUJ 需要的搜索时间	OFF : 读时间没有缩短 ON : 读时间缩短	<ul style="list-style-type: none"> 变为 ON 以缩短在 A8UPU/A8PUJ 中的搜索时间。(在这种情况下，扫描时间增加 10%) * A8UPU/A8PUJ 不能用在 QCPU/QnACPU 特殊继电器中。 	○
M9081	SM1081	SM714	通信请求注册区域 BUSY 信号	OFF : 通信请求注册区域中有空闲空间 ON : 通信请求注册区域无空闲空间	<ul style="list-style-type: none"> 连接到 AJ71PT32-S3, A2C 或 A52G 的远程终端模块的通信许可/禁止指示。 	QnA
M9084	SM1084		错误检查	OFF : 执行错误检查 ON : 无错误检查	<ul style="list-style-type: none"> 当 END 指令被处理时，此继电器被用于设置是否执行错误检查(用于设置 END 指令处理时间)。 保险丝破损检测 电池检查 I/O 模块校对检查 	○
M9091	SM1091		指令错误标志	OFF : 无错误 ON : 有错误	<ul style="list-style-type: none"> 操作错误的详细因素被存储到 SD1091 时，变为 ON。 如果从那以后条件恢复为正常，保持为 ON。 	
M9094	SM1094	SM251	I/O 改变标志	OFF : 替换 ON : 无替换	<ul style="list-style-type: none"> 在被改变的 I/O 模块的头 I/O 号码设置到 SD251 后，当 SM251 切换为 ON 时，这些 I/O 模块可以在线改变。(一次设置仅允许更改一个模块。) 在程序或外围设备测试模式中切换到 ON，以改变处于 CPU RUN 中的模块。在外围设备测试模式中切换为 ON，以改变处于 CPU STOP 中的模块。 RUN/STOP 模式在 I/O 模块改变完成前不能被改变。 	QnA
M9100	SM1100	SM320	存在/不存在 SFC 程序	OFF : SFC 程序未被使用 ON : SFC 程序被使用	<ul style="list-style-type: none"> 如果 SFC 程序已登记则变为 ON，并且在 SFC 未登记时变为 OFF。 	○
M9101	SM1101	SM321	启动/停止 SFC 程序	OFF : SFC 程序停止 ON : SFC 程序启动	<ul style="list-style-type: none"> 如果 SFC 程序将被启动，应该由程序将其变为 ON。如果变为 OFF，则执行步的操作输出变为 OFF 并且 SFC 程序停止。 	
M9102	SM1102	SM322	SFC 程序启动状态	OFF : 初始化启动 ON : 继续	<ul style="list-style-type: none"> 当使用 SM322 重新启动 SFC 程序时，选择启动步。ON: 当 SFC 程序停止时所有执行状态都被清除，并且程序从块 0 的初始步处重新启动。OFF: 当程序停止时，从正被执行块的程序步开始。 一旦变为 ON，程序被锁在系统中，并且即使系统断电也保持为 ON。系统上电后或者从块 0 的初始步启动时，应该由顺序程序切换为 OFF。 	

特殊继电器列表 (续表)

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义			详细信息	可用的 CPU
M9103	SM1103	SM323	存在/不存在连续转换	OFF : 连续转换无效 ON : 连续转换有效			● 当所有连续步的转换条件都建立时, 对转换条件已经建立的步, 选择是连续转换还是一步一步的转换。 ON: 执行连续转换。 OFF: 每个扫描周期转换一步。	○
M9104	SM1104	SM324	连续转换挂起标志	OFF : 当转换完成时 ON : 当无转换时			● 在操作处于连续转换模式或连续转换时为 OFF, 当没有执行连续转换时为 ON。 ● 在没有连续转换模式的操作中常为 ON。	
M9108	SM1108	SM90	步转换看门狗定时器启动(和 D9108 等价)	OFF : 看门狗定时器复位 ON : 看门狗定时器复位启动			● 当步转换看门狗定时器的测量被启动时变为 ON。 将此继电器切换为 OFF, 以复位步转换看门狗定时器。	
M9109	SM1109	SM91	步转换看门狗定时器启动(和 D9109 等价)					
M9110	SM1110	SM92	步转换看门狗定时器启动(和 D9110 等价)					
M9111	SM1111	SM93	步转换看门狗定时器启动(和 D9111 等价)					
M9112	SM1112	SM94	步转换看门狗定时器启动(和 D9112 等价)					
M9113	SM1113	SM95	步转换看门狗定时器启动(和 D9113 等价)					
M9114	SM1114	SM96	步转换看门狗定时器启动(和 D9114 等价)					
M9180	SM1180	SM825	激活步采样跟踪完成标志	OFF : 跟踪启动 ON : 跟踪完成			● 当所有指定块的采样跟踪完成时置位。当采样跟踪启动时复位。	
M9181	SM1181	SM822	激活步采样跟踪执行标志	OFF : 跟踪未被执行 ON : 跟踪正在执行			● 当采样跟踪正被执行时置位。 当采样跟踪已完成或挂起时复位。	
M9182	SM1182	SM821	激活步采样跟踪允许	OFF : 跟踪禁止/挂起 ON : 跟踪许可			● 选择采样跟踪执行许可/禁止。 ON: 采样跟踪执行被许可。 OFF: 采样跟踪执行被禁止。 如果在采样跟踪执行中变为 OFF, 则跟踪被挂起。	
M9196	SM1196	SM325	在块停止时的操作输出	OFF : 线圈输出 OFF ON : 线圈输出 ON			● 当块停止被执行时选择操作输出。 ON: 通过使用在块停止时被执行步的操作输出, 保持被使用线圈的 ON/OFF 状态。 OFF: 所有线圈输出都变为 OFF。(SET 指令的操作输出的被保持, 而不管 M9196 的 ON/OFF 状态)	
M9197	SM1197		在烧坏的保险丝和 I/O 校验错误显示之间切换	SM9197	SM1198	即将显示的 I/O 号	根据 SM1197 和 SM1198 中 ON/OFF 状态的组合, 在保险丝烧坏模块存储寄存器(SD1100 到 SD1107)中的 I/O 号码和 I/O 模块校验错误存储寄存器(SD1116 到 SD1123)中的 I/O 号码之间切换。	
M9198	SM1198				OFF	OFF		X/Y 0 到 7F0
		ON			OFF	X/Y 800 到 FF0		
		OFF			ON	X/Y 1000 到 17F0		
				ON	ON	X/Y 1800 到 1FF0		

特殊继电器列表 (续表)

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义	详细信息	可用的 CPU
M9199	SM1199		在线采样跟踪/状态闭锁的数据恢复	OFF : 数据恢复禁止 ON : 数据恢复允许	<ul style="list-style-type: none"> 当采样跟踪/状态闭锁被执行时, 在重新启动时恢复 CPU 模块中的设置数据。 为了重新执行, SM1199 应为 ON。(当从外围设备重新写数据时则没有必要) 	○
M9200	SM1200	—	ZNRD 指令 (LRDP 指令, 用于 ACPU)接收(用于主站)	OFF : 不被接收 ON : 被接收	<ul style="list-style-type: none"> 取决于 ZNRD (字设备读)指令是否被接收。 在程序中用作 ZNRD 的互锁指令。 使用 RST 指令进行复位。 	QnA
M9201	SM1201	—	ZNRD 指令 (LRDP 指令, 用于 ACPU)完成(用于主站)	OFF : 未完成 ON : 结束	<ul style="list-style-type: none"> 取决于 ZNRD (字设备读)指令的执行是否已完成。 在 ZNRD 指令执行完成后, 用作复位 M9200 和 M9201 的条件触点。 使用 RST 指令进行复位。 	
M9202	SM1202	—	ZNWR 指令 (LWTP 指令, 用于 ACPU)接收(用于主站)	OFF : 不被接收 ON : 接收	<ul style="list-style-type: none"> 取决于 ZNWR (字设备写)指令是否被接收。 在程序中用作 ZNWR 指令的互锁。 使用 RST 指令进行复位。 	
M9203	SM1203	—	ZNWR 指令 (LWTP 指令, 用于 ACPU)完成(用于主站)	OFF : 未完成 ON : 结束	<ul style="list-style-type: none"> 取决于 ZNWR (字设备写)指令执行是否已经完成。 用作在 ZNWR 指令完成后复位 M9202 和 M9203 的条件触点。 使用 RST 指令进行复位。 	
M9204	SM1204	—	ZNRD 指令 (LRDP 指令, 用于 ACPU)接收(用于本地站)	OFF : 未完成 ON : 结束	表明 ZNRD 指令是在本地站完成的。	
M9205	SM1205	—	ZNWR 指令 (LWTP 指令, 用于 ACPU)接收(用于本地站)	OFF : 未完成 ON : 结束	表明 ZNWR 指令是在本地站完成的。	
M9206	SM1206	—	主机站链接参数错误	OFF : 正常 ON : 不正常	取决于主站的链接参数设置是否有效。	
M9207	SM1207	—	链接参数检查结果	OFF : YES ON : NO	取决于在三级系统中, 第二级主站的链接参数设置是否和第三级主站的链接参数设置匹配。 (只对三级系统中的主站有效。)	
M9208	SM1208	—	设定主站 B 和 W 的传送范围(只对下级链接主站)	OFF : 传送到第 2 级和第 3 级 ON : 只传送到第 2 级	<ul style="list-style-type: none"> 取决于由高级—链接主站(自站)控制的 B 和 W 数据是否被送往低级—链接的本地站(第三级站) <ul style="list-style-type: none"> 当 SM1208 为 OFF 时.... 主机站的 B 和 W 被送往第三级站。 当 SM1208 为 ON 时..... 主机站的 B 和 W 不被送往第三级站。 	
M9209	SM1209	—	链接参数检查命令(只对下级链接主站)	OFF : 执行检查功能 ON : 检查不执行	<ul style="list-style-type: none"> 设置为 ON, 以便不匹配上级和下级链接的 B 和 W。 当 SM1209 为 ON 时, 上级和下级链接的链接参数不被检查。 当 SM1209 为 OFF 时, 上级和下级链接的链接参数被检查。 	
M9210	SM1210	—	链接卡错误(用于主站)	OFF : 正常 ON : 不正常	控制是否被执行取决于链接卡硬件是否有故障。	
M9211	SM1211	—	链接模块错误(用于本地站)	OFF : 正常 ON : 不正常	控制是否被执行取决于链接卡硬件是否有故障。	
M9224	SM1224	—	链接状态	OFF : 在线 ON : 离线, 站—站测试, 或自—回路返回测试	取决于主站是在线还是离线, 是处于站—站测试模式还是处于自—回路返回测试模式。	
M9225	SM1225	—	前向回路错误	OFF : 正常 ON : 不正常	取决于前向回路线路的错误条件。	
M9226	SM1226	—	反向回路错误	OFF : 正常 ON : 不正常	取决于反向回路线路的错误条件。	

特殊继电器列表 (续表)

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义	详细信息	可用的 CPU
M9227	SM1227	—	回路测试状态	OFF : 未被执行 ON : 前向或反向回路检测执行正在进行	取决于主站是否正在执行前向或反向回路测试。	QnA
M9232	SM1232	—	本地站操作状态	OFF : RUN 或 STEP RUN 状态 ON : STOP 或 PAUSE 状态	控制是否执行取决于本地站是否处于 STOP 或 PAUSE 模式。	
M9233	SM1233	—	本地站错误检测状态	OFF : 无错误 ON : 错误检测	取决于本地站是否已从其它站检测到错误。	
M9235	SM1235	—	本地站, 远程 I/O 站参数错误检测状态	OFF : 无错误 ON : 错误检测	取决于本地或者远程 I/O 站是否在主站中检测到任何链接参数错误。	
M9236	SM1236	—	本地站, 远程 I/O 站初始通信状态	OFF : 无通信 ON : 通信正在进行	取决于本地或远程 I/O 站和主站之间的初始化通信结果。(参数通信, 等)	
M9237	SM1237	—	本地站, 远程 I/O 站错误	OFF : 正常 ON : 不正常	取决于本地或远程 I/O 站的错误条件。	
M9238	SM1238	—	本地站, 远程 I/O 站前向或反向回路错误	OFF : 正常 ON : 不正常	取决于本地或远程 I/O 站前向和反向回路线路的错误条件。	
M9240	SM1240	—	链接状态	OFF : 在线 ON : 离线, 站-到-站测试, 或自回路返回测试	取决于本地站是在线还是离线, 或者是在站-到-站测试还是自回路返回测试模式	
M9241	SM1241	—	前向回路线路错误	OFF : 正常 ON : 不正常	取决于前向回路线路的错误条件。	
M9242	SM1242	—	反向回路线路错误	OFF : 正常 ON : 不正常	取决于反向回路线路的错误条件。	
M9243	SM1243	—	回路返回执行	OFF : 回路返回未被执行 ON : 回路返回执行	取决于在本地站是否发生回路返回。	
M9246	SM1246	—	数据未被接收	OFF : 接收 ON : 未被接收	取决于从主站来的数据是否被接收。	
M9247	SM1247	—	数据未被接收	OFF : 接收 ON : 未被接收	取决于在三级系统中一个第三级站是否已从它的主站接收到数据。	
M9250	SM1250	—	参数未被接收	OFF : 接收 ON : 未被接收	取决于是否已经从主站接收到链接参数。	
M9251	SM1251	—	链接继电器	OFF : 正常 ON : 不正常	取决于本地站的数据链接条件。	
M9252	SM1252	—	回路测试状态	OFF : 未被执行 ON : 前向或反向回路测试执行正在进行中	取决于本地站是否正在执行前向或反向回路测试。	
M9253	SM1253	—	主站操作状态	OFF : RUN 或 STEP RUN 状态 ON : STOP 或 PAUSE 状态	控制是否被执行取决于主站是否处于 STOP 或 PAUSE 模式。	
M9254	SM1254	—	非主站的本地站操作状态	OFF : RUN 或 STEP RUN 状态 ON : STOP 或 PAUSE 状态	控制是否被执行取决于非主站的本地站是否处于 STOP 或 PAUSE 模式。	
M9255	SM1255	—	非主站的本地站错误	OFF : 正常 ON : 不正常	控制是否被执行取决于非主机站的本地站是否出错。	

特殊继电器列表

(10)用于冗余系统(主系统 CPU 信息*1)，只用于 Q4AR。

SM1510 到 SM1599 只对冗余系统有效。对独立系统都为 OFF。

号码	名称	含义	解释	设定者 (当被设定时)	ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM1500	保持模式	OFF : 无保持 ON : 保持	● 对 S.IN 指令范围检测发生范围溢出时，指定是否保持输出值。	U	新增	Q4AR
SM1501	保持模式	OFF : 无保持 ON : 保持	● 对 S.OUT 指令范围检测发生范围溢出时，指定是否保持输出值。	U	新增	
SM1510	操作模式	OFF : 冗余系统备份模式，独立系统 ON : 冗余系统分离模式	● 当操作模式为冗余系统分离时，变为 ON。	S (每个 END)	新增	
SM1511	当供电电源为 ON 时，启动模式	OFF : 系统 A 固定模式 ON : 前一控制系统闭锁模式	● 在系统上电时，当一个冗余系统的启动模式是前一控制系统闭锁模式时，变为 ON。	S(初始化)	新增	
SM1512	当 CPU 启动时，启动模式	OFF : 初始化启动 ON : 热启动	● 当冗余系统启动时，如果 CPU 模块的操作模式为热启动，则变为 ON。	S(初始化)	新增	
SM1513	操作模式，当 CPU 启动时	OFF : 初始化启动 ON : 热启动	● 当冗余系统实际启动时，如果 CPU 模块的操作模式为热启动，则变为 ON。	S(初始化)	新增	
SM1514	操作模式，当 CPU 切换时	OFF : 初始化启动 ON : 热启动	● 当 CPU 模块操作为冗余系统切换时，如果操作为热启动，则变为 ON。	S(初始化)	新增	
SM1515	输出保持模式	OFF : 输出复位 ON : 输出保持	● 当在停止错误中输出模式为输出保持时，变为 ON。	S (每个 END)	新增	
SM1516	操作系统状态	OFF : 控制系统 ON : 备用系统	● 当 CPU 模块操作系统状态为备用系统时，变为 ON。	S (状态改变)	新增	
SM1517	CPU 启动状态	OFF : 供电电源为 ON 启动 ON : 操作系统开关启动	● 当 CPU 模块是由操作系统开关启动，变为 ON。 ● 使用用户程序进行复位。	S (状态改变) /U	新增	
SM1518	跟踪执行模式	OFF : 批操作模式 ON : 进位溢出模式	● 当此继电器变为 OFF，如果在 END 处跟踪存储器正被使用，则跟踪启动将被延迟到此继电器为可执行。 ● 当此继电器变为 ON，如果在 END 处跟踪存储器正被使用，此跟踪的启动将被进位到下一个 END。	U	新增	
SM1520	数据跟踪传送链接指定	OFF : 无触发 ON : 有触发	SM1520 块 1	● 当数据被数据跟踪指令传送时，将数据块指定到触发器。 S. TRUCK.	U	新增
SM1521			SM1521 块 2			
SM1522			SM1522 块 3			
SM1523			SM1523 块 4			
SM1524			SM1524 块 5			
SM1525			SM1525 块 6			
SM1526			SM1526 块 7			
SM1527			SM1527 块 8			
SM1528			SM1528 块 9			
SM1529			SM1529 块 10			
SM1530			SM1530 块 11			
SM1531			SM1531 块 12			
SM1532			SM1532 块 13			
SM1533			SM1533 块 14			
SM1534			SM1534 块 15			
SM1535			SM1535 块 16			
SM1536			SM1536 块 17			
SM1537			SM1537 块 18			
SM1538			SM1538 块 19			
SM1539			SM1539 块 20			
SM1540			SM1540 块 21			
SM1541			SM1541 块 22			
SM1542			SM1542 块 23			
SM1543			SM1543 块 24			
SM1544			SM1544 块 25			
SM1545			SM1545 块 26			

特殊继电器列表 (续表)

号码	名称	含义	解释			设定者 (当被设定时)	ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM1546	数据跟踪传送连接指定	OFF : 无触发器 ON : 有触发器	SM1546	块 27	● 当数据被数据跟踪指令发送时， 将数据块指定到触发器。 S. TRUCK.	U	新增	Q4AR
SM1547			SM1547	块 28				
SM1548			SM1548	块 29				
SM1549			SM1549	块 30				
SM1550			SM1550	块 31				
SM1551			SM1551	块 32				
SM1552			SM1552	块 33				
SM1553			SM1553	块 34				
SM1554			SM1554	块 35				
SM1555			SM1555	块 36				
SM1556			SM1556	块 37				
SM1557			SM1557	块 38				
SM1558			SM1558	块 39				
SM1559			SM1559	块 40				
SM1560			SM1560	块 41				
SM1561			SM1561	块 42				
SM1562			SM1562	块 43				
SM1563			SM1563	块 44				
SM1564			SM1564	块 45				
SM1565			SM1565	块 46				
SM1566			SM1566	块 47				
SM1567			SM1567	块 48				
SM1568			SM1568	块 49				
SM1569			SM1569	块 50				
SM1570			SM1570	块 51				
SM1571			SM1571	块 52				
SM1572			SM1572	块 53				
SM1573			SM1573	块 54				
SM1574			SM1574	块 55				
SM1575			SM1575	块 56				
SM1576			SM1576	块 57				
SM1577			SM1577	块 58				
SM1578			SM1578	块 59				
SM1579			SM1579	块 60				
SM1580			SM1580	块 61				
SM1581			SM1581	块 62				
SM1582			SM1582	块 63				
SM1583			SM1583	块 64				
SM1590	来自网络模块的切换状态	OFF : 正常 ON : 切换不成功	● 如果网络模块检测到一个网络故障并且已往主 CPU 模块发送了切换请求，当切换不能被正常执行时，变为 ON。			S (发生错误)		

特殊继电器列表

(11)用于冗余系统(其它系统 CPU 信息*1)，只用于 Q4AR

SM1600 到 SM1650 只对 CPU 冗余系统备份模式有效。因此它们不能在分离模式中被刷新。

无论是备份模式还是分离模式对 SM4651 到 SM1699 都有效。

SM1600 到 SM1699 对独立系统都变为 OFF。

号码	名称	含义	解释	设定者 (当被设定时)	ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> * 2	可用的 CPU
SM1600	诊断错误	OFF : 无错误 ON : 有错误	● 如果在诊断结果里面有错误发生, 则变为 ON。 (包括外部诊断) ● 保持 ON, 即使从那以后变为正常。	S (每个 END)	新增	Q4AR
SM1601	自诊断错误	OFF : 无自诊断错误 ON : 自诊断错误	● 如果在自诊断结果里面有错误发生, 则变为 ON。 ● 保持 ON, 即使从那以后变为正常。	S (每个 END)	新增	
SM1605	错误公用信息	OFF : 无错误公用信息 ON : 有错误公用信息	● 当有错误公用信息并且 SM1600 为 ON 时, 变为 ON。	S (每个 END)	新增	
SM1616	错误单独信息	OFF : 无错误单独信息 ON : 有错误单独信息	● 当有错误单独信息并且 SM1600 为 ON 时, 变为 ON。	S (每个 END)	新增	
SM1653	STOP 触点	STOP 状态	● 当处于 STOP 状态时, 变为 ON。	S (每个 END)	新增	
SM1654	PAUSE 触点	PAUSE 状态	● 当处于 PAUSE 状态时, 变为 ON。	S (每个 END)	新增	
SM1655	STEP-RUN 触点	STEP-RUN 状态	● 当处于 STEP-RUN 状态时, 变为 ON。	S (每个 END)	新增	

* 1 存储其它系统 CPU 诊断信息和系统信息。

* 2 这表示用于主系统 CPU 的特殊继电器(SM ☐ ☐)。

(12)用于冗余系统(跟踪)，只用于 Q4AR。

无论是备用模式还是第二模式都对 SM1700 到 SM1799 有效。对单独系统这些继电器都变为 OFF。

号码	名称	含义	解释			设定者 (当被设定时)	ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM1700	跟踪执行标志	OFF : 执行不可能 ON : 执行可能	● 当跟踪被正常执行时变为 ON。			S (状态改变)	新增	Q4AR
SM1712	传送触发结束标志	OFF : 传送未完成 ON : 传送结束	SM1712	块 1	● 当相应的数据传送完成时一个扫描周期为 ON。	S (状态改变)	新增	
SM1713			SM1713	块 2				
SM1714			SM1714	块 3				
SM1715			SM1715	块 4				
SM1716			SM1716	块 5				
SM1717			SM1717	块 6				
SM1718			SM1718	块 7				
SM1719			SM1719	块 8				
SM1720			SM1720	块 9				
SM1721			SM1721	块 10				
SM1722			SM1722	块 11				
SM1723			SM1723	块 12				
SM1724			SM1724	块 13				
SM1725			SM1725	块 14				
SM1726			SM1726	块 15				
SM1727			SM1727	块 16				
SM1728			SM1728	块 17				
SM1729			SM1729	块 18				
SM1730			SM1730	块 19				
SM1731			SM1731	块 20				
SM1732			SM1732	块 21				
SM1733			SM1733	块 22				

特殊继电器列表 (续表)

号码	名称	含义	解释			设定者 (当被设定时)	ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM1734	传送触发结束标志	OFF : 传送未完成 ON : 传送结束	SM1734	块 23	● 当相应的数据传送完成时一个扫描周期为 ON。	S (状态改变)	新增	Q4AR
SM1735			SM1735	块 24				
SM1736			SM1736	块 25				
SM1737			SM1737	块 26				
SM1738			SM1738	块 27				
SM1739			SM1739	块 28				
SM1740			SM1740	块 29				
SM1741			SM1741	块 30				
SM1742			SM1742	块 31				
SM1743			SM1743	块 32				
SM1744			SM1744	块 33				
SM1745			SM1745	块 34				
SM1746			SM1746	块 35				
SM1747			SM1747	块 36				
SM1748			SM1748	块 37				
SM1749			SM1749	块 38				
SM1750			SM1750	块 39				
SM1751			SM1751	块 40				
SM1752			SM1752	块 41				
SM1753			SM1753	块 42				
SM1754			SM1754	块 43				
SM1755			SM1755	块 44				
SM1756			SM1756	块 45				
SM1757			SM1757	块 46				
SM1758			SM1758	块 47				
SM1759			SM1759	块 48				
SM1760			SM1760	块 49				
SM1761			SM1761	块 50				
SM1762			SM1762	块 51				
SM1763			SM1763	块 52				
SM1764			SM1764	块 53				
SM1765			SM1765	块 54				
SM1766			SM1766	块 55				
SM1767			SM1767	块 56				
SM1768			SM1768	块 57				
SM1769			SM1769	块 58				
SM1770			SM1770	块 59				
SM1771			SM1771	块 60				
SM1772			SM1772	块 61				
SM1773			SM1773	块 62				
SM1774			SM1774	块 63				
SM1775			SM1775	块 64				

附录 3.3 过程控制 CPU 的特殊继电器列表

特殊继电器，SM，是应用固化在 PLC 中的内部继电器。
 由于这个原因，它们不能象普通的内部继电器那样被顺控程序使用。
 然而，如果需要它们可以变为 ON 或 OFF 以控制 CPU 模块和远程 I/O 模块。

下表中标题的含义如下所示。

项目	项目功能
号码	● 表示特殊继电器的号码。
名称	● 表示特殊继电器的名称。
含义	● 表示特殊继电器的种类。
解释	● 包含关于特殊继电器内容的详细信息。
设置者(当被设置时)	● 表示继电器是有系统设置还是由用户设置，并且，如果是由系统设置，当设置被执行时。 <设置者> S : 由系统设置 U : 由用户设置(在 GX Developer 上的顺序程序或测试操作中) S/U : 由系统和用户共同设置 <当被设置时> → 表示只由系统设置 每个 END : 在每个 END 处理过程中设置 初始化 : 只在初始化处理过程中设置 (当电源变为 ON, 或者当从 STOP 变为 RUN 时) 状态改变 : 在有状态改变时设置 错误 : 当有错误产生时设置 指令执行 : 当指令被执行时设置 请求 : 当有用户请求时设置(通过 SM, 等)
相应的 ACPU M9 □ □ □	● 表示特殊继电器 M9 □ □ □ 和 ACPU 相对应。 (当注释中发生变化时改变和符号) ● “新增”表示此项目是新近增加到高性能模型 QCPU/QnACPU。
相应的 CPU	● 表示相应的 CPU 模块类型名称。 ○+Rem: 可以被应用到所有的 CPU 类型和 MELSECNET/H 远程 I/O 模块。 ○: 可以被应用到所有类型的 CPU 远程: 可以应用到 MELSECNET/H 远程 I/O 模块。

对于下列项目的详细信息，参见下列手册：

- 网络 → ● 用于 Q 相应的 MELSECNET/H 网络系统参考手册(PLC 到 PLC 网络)
- 用于 Q 相应的 MELSECNET/H 网络系统参考手册(远程 I/O 网络)
- SFC → QCPU(Q 模式)/QnACPU 编程手册(SFC)

特殊继电器列表

(1) 诊断信息

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM0	诊断错误	OFF : 无错误 ON : 有错误	<ul style="list-style-type: none"> 如果作为诊断结果有错误发生则, 变为 ON。 (包括当一个报警器为 ON, 并且当有错误被 CHK 指令检测到) 如果 从那以后条件重新变为正常, 则保持 ON。 	S(错误)	新增	○+Rem
SM1	自诊断错误	OFF : 无自诊断错误 ON : 自诊断	<ul style="list-style-type: none"> 如果作为诊断结果有错误发生, 则变为 ON。(不包括当一个报警器为 ON 或当有错误被 CHK 指令检测到) 如果从那以后条件重新变为正常, 则保持 ON。 	S(错误)	M9008	
SM5	错误公用信息	OFF : 无错误公用信息 ON : 有错误公用信息	<ul style="list-style-type: none"> 当 SM0 为 ON 时, 如果有错误公用信息则为 ON。 	S(错误)	新增	
SM16	错误单独信息	OFF : 无错误公用信息 ON : 有错误公用信息	<ul style="list-style-type: none"> 当 SM0 为 ON 时, 如果有错误单独信息则为 ON。 	S(错误)	新增	
SM50	错误复位	OFF → ON : 错误复位	<ul style="list-style-type: none"> 执行错误复位操作 参见章节 11.3 以获得进一步信息 	U	新增	
SM51	电池低闭锁	OFF : 正常 ON : 电池低	<ul style="list-style-type: none"> 如果 CPU 模块或存储卡上的电池电压低于额定值, 则变为 ON。 如果电池电压重新变为正常, 则保持 ON。 同步于 BAT. LED 	S(错误)	M9007	○
SM52	电池低	OFF : 正常 ON : 电池低	<ul style="list-style-type: none"> 和 SM51 一样, 但是在电池电压重新变为正常后最终变为 OFF。 	S(错误)	M9006	
SM53	AC/DC DOWN 检测	OFF : AC/DC DOWN 不被检测 ON : AC/DC DOWN 被检测	<ul style="list-style-type: none"> 如果在使用 AC 电源供电模块时有 20ms 内的瞬时电源故障发生, 则变为 ON。 当电源切换到 OFF 时复位, 然后为 ON。 如果在使用 DC 电源供电模块时有 10ms 内的瞬时电源故障发生, 则变为 ON。 当电源切换到 OFF 时复位, 然后为 ON。 	S(错误)	M9005	
SM56	操作错误	OFF : 正常 ON : 操作错误	<ul style="list-style-type: none"> 当操作错误发生时变为 ON。 如果从那以后条件重新变为正常, 则保持 ON。 	S(错误)	M9011	
SM60	烧坏保险丝检测	OFF : 正常 ON : 模块有烧坏的保险丝	<ul style="list-style-type: none"> 如果至少有一个输出模块的保险丝被烧坏, 则变为 ON。 如果从那以后条件重新变为正常, 则保持 ON。 即使对远程 I/O 站的输出模块, 烧坏的保险丝状态也被检测。 	S(错误)	M9000	○+Rem
SM61	I/O 模块确认错误	OFF : 正常 ON : 有错误	<ul style="list-style-type: none"> 如果 I/O 模块和上电时指定的状态不同, 则变为 ON。 如果从那以后条件重新变为正常, 则保持 ON。 I/O 模块确认操作对远程 I/O 站模块也进行。 	S(错误)	M9002	
SM62	报警器检测	OFF : 不被检测 ON : 被检测	<ul style="list-style-type: none"> 如果即使只有一个报警器 F 变为 ON, 则变为 ON。 	S (指令执行)	M9009	
SM80	CHK 检测	OFF : 不被检测 ON : 被检测	<ul style="list-style-type: none"> 如果有错误被 CHK 指令检测到, 则变为 ON。 如果 从那以后条件重新变为正常, 则保持 ON。 	S (指令执行)	新增	○
SM90	用于步转换的看门狗定时器的启动 (只有在 SFC 程序存在时才被允许)	OFF : 未被启动 (看门狗定时器复位) ON : 启动 (看门狗定时器启动)	对应于 SD90	U	M9108	
SM91			对应于 SD91		M9109	
SM92			对应于 SD92		M9110	
SM93			对应于 SD93		M9111	
SM94			对应于 SD94		M9112	
SM95			对应于 SD95		M9113	
SM96			对应于 SD96		M9114	
SM97			对应于 SD97		新增	
SM98			对应于 SD98		新增	
SM99			对应于 SD99		新增	
			<ul style="list-style-type: none"> 当步转换看门狗定时器的测试开始时, 变为 ON。 当其变为 OFF 时, 复位步转换看门狗定时器。 			

特殊继电器列表

(2) 系统信息

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM202	LED OFF 命令	OFF →ON : LED 关闭	● 当此继电器从 OFF 变为 ON, 和 SD202 上每个位相对应的 LED 关闭。	U	新增	○
SM203	STOP 触点	STOP 状态	● 当处于 STOP 状态时, 变为 ON。	S (状态改变)	M9042	
SM204	PAUSE 触点	PAUSE 状态	● 当处于 PAUSE 状态时, 变为 ON。	S (状态改变)	M9041	
SM206	PAUSE 使能线圈	OFF : PAUSE 无效 ON : PAUSE 有效	● 如果当远程 PAUSE 触点变为 ON 时此继电器为 ON, 则进入 PAUSE 状态。	U	M9040	远程
	设备元件测试请求接受状态	OFF : 设备元件测试未被执行 ON : 设备元件测试被执行	● 当设备元件测试模式在 GX Developer 上被执行时, 变为 ON。	S(请求)	新增	
SM210	时钟数据设置请求	OFF : 忽略 ON : 设置请求	● 当此继电器从 OFF 变为 ON 时, 在改变扫描周期的 END 指令执行结束后存储在从 SD210 到 SD213 的时钟数据被写入时钟设备元件。	U	M9025	○
SM211	时钟数据错误	OFF : 无错误 ON : 错误	● 当在时钟数据(SD210 到 SD213)中产生错误时为 ON, 在没有错误被检测到时为 OFF。	S(请求)	M9026	
SM213	时钟数据读请求	OFF : 忽略 ON : 读请求	● 当此继电器为 ON 时, 时钟数据以 BCD 值的形式读到 SD210 和 SD213 中。	U	M9028	○+Rem
SM235	在线模块改变标志	OFF : 在线模块改变不在处理中 ON : 在线模块改变处理中	● 在在线模块改变过程中变为 ON。(用于主机 PLC)	S(在在线模块改变过程中)	新增	
SM236	在线模块改变完成标志	OFF : 在线模块改变未完成 ON : 在线模块改变完成	● 在线模块改变完成后, 变为 ON 一个扫描周期。 ● 触点只能被扫描程序使用。(用于主机 PLC)	S(当在线模块改变已经完成时)	新增	
SM240	第 1 CPU 复位标志	OFF : 第 1 CPU 复位取消 ON : 第 1 CPU 重新设置	● 当第 1CPU 复位被取消时, 变为 OFF。 ● 当第 1 CPU 被重新设置(包括 PLC 从基板中移掉的情况)时, 变为 ON。 ● 其它 PLC 也被置入复位状态。	S (状态改变)	新增	○
SM241	第 2 CPU 复位标志	OFF : 第 2 CPU 复位取消 ON : 第 2 CPU 重新设置	● 当第 2 CPU 的复位被取消时, 变为 OFF。 ● 当第 2 CPU 被重新设置(包括 PLC 从基板中移掉的情况)时, 变为 ON。 ● 其它 PLC 进入“MULTI CPU DOWN”(错误代码: 7000)。			
SM242	第 3 CPU 复位标志	OFF : 第 3 CPU 复位取消 ON : 第 3 CPU 重新设置	● 当第 3 CPU 的复位被取消时, 变为 OFF。 ● 当第 3 CPU 被重新设置(包括 PLC 从基板中移掉的情况)时, 变为 ON。 ● 其它 PLC 进入“MULTI CPU DOWN”(错误代码: 7000)。			
SM243	第 4 CPU 复位标志	OFF : 第 4 CPU 复位取消 ON : 第 4 CPU 重新设置	● 当第 4 CPU 的复位被取消时, 变为 OFF。 ● 当第 4 CPU 被重新设置(包括 PLC 从基板中移掉的情况)时, 变为 ON。 ● 其它 PLC 进入“MULTI CPU DOWN”(错误代码: 7000)。			
SM244	第 1 CPU 错误标志	OFF : 第 1 CPU 正常 ON : 第 1 CPU 处于停止错误	● 当第 1 CPU 为正常(包含连续错误)时, 变为 OFF。 ● 当第 1 CPU 处于停止错误时, 变为 ON。			
SM245	第 2 CPU 错误标志	OFF : 第 2 CPU 正常 ON : 第 2 CPU 处于停止错误	● 当第 2 CPU 正常(包含连续错误)时, 变为 OFF。 ● 当第 2 CPU 处于停止错误时, 变为 ON。			
SM246	第 3 CPU 错误标志	OFF : 第 3 CPU 正常 ON : 第 3 CPU 处于停止错误	● 当第 3 CPU 正常时, 变为 OFF。(包含连续错误) ● 当第 3 CPU 处于停止错误时, 变为 ON。			
SM247	第 4 CPU 错误标志	OFF : 第 4 CPU 正常 ON : 第 4 CPU 处于停止错误	● 当第 4 CPU 正常时, 变为 OFF。(包含连续错误) ● 当第 4 CPU 处于停止错误时, 变为 ON。			
SM250	最大载入的 I/O 读	OFF : 忽略 ON : 读	● 当此继电器从 OFF 变为 ON 时, 将安装 I/O 号码的最大值读到 SD250。	U	新增	○+Rem
SM254	所有站刷新命令	OFF : 刷新访问的站 ON : 刷新所有站	● 使批量刷新有效(对低速周期也有效)。 ● 指定是只接收到站的站还是接收所有从站。	U (每个 END)	新增	○

特殊继电器列表 (续表)

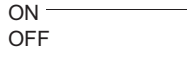



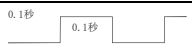

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM255	MELSECNET/10 模块 1 信息	OFF : 活动网络 ON : 备用网络	● 为备用网络变为 ON (如果没有关于活动网络或备用网络的指定, 则假设为活动网络)	S(初始化)	新增	○
SM256		OFF : 读 ON : 不读	● 对从链接到 CPU 模块的刷新(B、W 等), 表示是否从链接模块中读取数据。	U	新增	
SM257		OFF : 写 ON : 不写	● 对从 CPU 模块到链接的刷新(B、W 等), 表示是否向链接模块中写入数据。	U	新增	
SM260	MELSECNET/10 模块 2 信息	OFF : 活动网络 ON : 备用网络	● 为备用网络变为 ON (如果没有关于活动网络或备用网络的指定, 则假设为活动网络)	S(初始化)	新增	
SM261		OFF : 读 ON : 不读	● 对从链接到 CPU 模块的刷新(B、W 等), 表示是否从链接模块中读取数据。	U	新增	
SM262		OFF : 写 ON : 不写	● 对从 CPU 模块到链接的刷新(B、W 等), 表示是否向链接模块中写入数据。	U	新增	
SM265	MELSECNET/10 模块 3 信息	OFF : 活动网络 ON : 备用网络	● 为备用网络变为 ON (如果没有关于活动网络或备用网络的指定, 则假设为活动网络)	S(初始化)	新增	
SM266		OFF : 读 ON : 不读	● 对从链接到 CPU 模块的刷新(B、W 等), 表示是否从链接模块中读取数据。	U	新增	
SM267		OFF : 写 ON : 不写	● 对从 CPU 模块到链接的刷新(B、W 等), 表示是否向链接模块中写入数据。	U	新增	
SM270	MELSECNET/10 模块 4 信息	OFF : 活动网络 ON : 备用网络	● 为备用网络变为 ON (如果没有关于活动网络或备用网络的指定, 则假设为活动网络)	S(初始化)	新增	○ + Rem
SM271		OFF : 读 ON : 不读	● 对从链接到 CPU 模块的刷新(B、W 等), 表示是否从链接模块中读取数据。	U	新增	
SM272		OFF : 写 ON : 不写	● 对从 CPU 模块到链接的刷新(B、W 等), 表示是否向链接模块中写入数据。	U	新增	
SM280	CC-Link 错误	OFF : 正常 ON : 错误	● 当在安装任一的 QJ61QBT11 中检测到 CC-Link 错误时变为 ON。当正常操作被恢复时变为 OFF。	S (状态改变)	新增	
SM320	存在/不存在 SFC 程序	OFF : SFC 程序不存在 ON : SFC 程序存在	● 当有 SFC 程序登记时, 变为 ON。 ● 当 SFC 程序没有登记时, 变为 OFF。	S(初始化)	M9100	
SM321	启动/停止 SFC 程序	OFF : SFC 程序停止 ON : SFC 程序启动	● 初始值设置为和 SM320 的值相同。 (如果存在 SFC 程序则自动变为 ON。) ● 当此继电器从 OFF 变为 ON, 则启动 SFC 程序。 ● 当此继电器从 ON 变为 OFF, 则停止 SFC 程序。	S(初始化) U	M9101 格式改变	
SM322	SFC 程序启动状态	OFF : 初始启动 ON : 重新启动	● 在 PLC 参数对话框的 SFC 设置中的 SFC 程序启动模式被设置为初始值。 在初始启动时: OFF 在连续启动时: ON	S(初始化) U	M9102 格式改变	○
SM323	存在/不存在对整个块的连续转换	OFF : 连续转换无效 ON : 连续转换有效	对 SFC 数据设备元件的“连续转换位”没有设置的块进行存在/不存在连续转换设置。	U	M9103	
SM324	连续转换防止标志	OFF : 当转换被执行时 ON : 当没有转换时	● 当处于连续转换模式操作中或处于连续转换时为 OFF, 当连续转换不被执行时为 ON。 ● 在无连续转换模式的操作中为常 ON。	S (指令执行)	M9104	
SM325	块停止时的输出模式	OFF : OFF ON : 保持	在块停止时选择是否保持活动步的线圈输出。 ● 作为初始值, 当线圈输出为 OFF 时在参数中停止的块的输出模式为 OFF, 当线圈输出被保持时为 ON。 ● 当此继电器为 OFF 时, 所有线圈输出变为 OFF。 ● 当此继电器为 ON 时, 线圈输出被保持。	S(初始化) U	M9196	

特殊继电器列表 (续表)

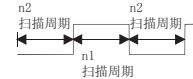
号码	名称	名称	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM326	SFC 设备清除模式	OFF : 清除设备 ON : 保持设备	在从 STOP 到程序写到 RUN 的切换时刻, 选择设备状态。 (除步进继电器外的所有设备)	U	新增	○
SM327	结束步执行时的输出	OFF : OFF ON : 保持	<ul style="list-style-type: none"> 当此继电器变为 OFF 时, 当转换建立时被保持的步 (SC, SE, ST) 的线圈输出在结束步到达时变为 OFF。 	S (初始化) U	新增	
SM330	用于低速执行类型程序的操作模式	OFF : 异步模式 ON : 同步模式	<ul style="list-style-type: none"> 选择低速执行类型程序是按照异步模式还是同步模式执行。 异步模式 (此继电器被变为 OFF。) 在额外时间内, 低速执行类型程序的操作被连续执行的模式。 同步模式 (此继电器被变为 ON。) 即使当存在额外时间时, 低速执行类型程序的操作从下一个扫描周期开始执行的模式。 	U	新增	
SM390	访问执行标志	ON 表示对智能功能模块访问的完成	<ul style="list-style-type: none"> 存储刚刚被执行的智能功能模块访问指令的状态。 (在智能功能模块访问指令被再次执行时覆盖此信息。) 此标志被用户在程序中用作完成位。 	S (状态改变)	新增	

(3) 系统时钟/计数器

特殊继电器列表

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM400	常 ON	ON  OFF	● 正常为 ON	S (每个 END 处理)	M9036	○
SM401	常 OFF	ON  OFF	● 正常为 OFF	S (每个 END 处理)	M9037	
SM402	只在 RUN 后 ON 一个扫描周期	ON  OFF	<ul style="list-style-type: none"> 在 RUN 后, ON 一个扫描周期 此连接只能被用于扫描周期执行类型程序。 当使用初始化执行类型程序时, 在 RUN 后的第一个扫描周期内, 在扫描执行类型程序的 END 处理时刻变为 OFF。 	S (每个 END 处理)	M9038	
SM403	在 RUN 后, 只 OFF 一个扫描周期	ON  OFF	<ul style="list-style-type: none"> 在 RUN 后, 只 OFF 一个扫描周期。 此连接只能被用于扫描周期执行类型程序。 当使用初始化执行类型程序时, 在 RUN 后的第一个扫描周期内, 在扫描执行类型程序的 END 处理时刻变为 ON。 	S (每个 END 处理)	M9039	
SM404	低速执行类型程序 在 RUN 后, ON 一个扫描周期	ON  OFF	<ul style="list-style-type: none"> 在 RUN 后, ON 一个扫描周期。 此连接只能用于低速执行类型程序。 	S (每个 END 处理)	新增	
SM405	低速执行类型程序 在 RUN 后, 只 OFF 一个扫描周期	ON  OFF	<ul style="list-style-type: none"> 在 RUN 后, 只 OFF 一个扫描周期。 此连接只能被用于低速执行类型程序 	S (每个 END 处理)	新增	
SM409	0.01 秒时钟		<ul style="list-style-type: none"> 以 5ms 为间隔在 ON 和 OFF 之间重复改变。 当 PLC 供电电源变为 OFF 或 CPU 模块被复位时, 从 OFF 开始启动。 注意, 在程序的执行过程中当指定的时间用完时 ON-OFF 状态改变。 	S (状态改变)	新增	
SM410	0.1 秒时钟		<ul style="list-style-type: none"> 在每个指定的时间间隔重复的在 ON 和 OFF 之间改变。 当 PLC 供电电源变为 OFF 或有 CPU 模块被复位时, 从 OFF 开始启动。 注意, 在程序的执行过程中当指定的时间用完时 ON-OFF 状态改变。 	S (状态改变)	M9030	
SM411	0.2 秒时钟				M9031	
SM412	1 秒时钟				M9032	
SM413	2 秒时钟				M9033	
SM414	2n 秒时钟		<ul style="list-style-type: none"> 此继电器按照在 SD414 中指定的时间 (单位: S) 间隔在 ON 和 OFF 之间交替改变。 当 PLC 供电电源变为 OFF 或有 CPU 模块被复位时, 从 OFF 开始启动。 注意, 在程序的执行过程中当指定的时间用完时 ON-OFF 状态改变。 	S (状态改变)	M9034 格式改变	
SM415	2n (ms) 时钟		<ul style="list-style-type: none"> 此继电器按照在 SD415 中指定的时间 (单位: ms) 间隔在 ON 和 OFF 之间交替改变。 当 PLC 供电电源变为 OFF 或有 CPU 模块被复位时, 从 OFF 开始启动。 注意, 在程序的执行过程中当指定的时间用完时 ON-OFF 状态改变。 	S (状态改变)	新增	

特殊继电器列表 (续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM420	0 号用户定时时钟		<ul style="list-style-type: none">继电器在固定扫描间隔重复 ON/OFF 切换。当 PLC 供电电源变为 OFF 或者 CPU 模块被复位, 从 OFF 开始启动。ON/OFF 间隔由 DUTY 指令设置。 <div><div>DUTY</div><div>n1</div><div>n2</div><div>SM420</div></div> <p>n1: ON 的扫描间隔 n2: OFF 的扫描间隔</p>	S (每个 END 处理)	M9020	○
SM421	1 号用户定时时钟				M9021	
SM422	2 号用户定时时钟				M9022	
SM423	3 号用户定时时钟				M9023	
SM424	4 号用户定时时钟				M9024	
SM430	5 号用户定时时钟		<ul style="list-style-type: none">用于 SM420 到 SM424 的低速执行类型程序	S (每个 END 处理)	新增	
SM431	6 号用户定时时钟					
SM432	7 号用户定时时钟					
SM433	8 号用户定时时钟					
SM434	9 号用户定时时钟					

(4) 扫描信息

特殊继电器列表

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM510	低速程序执行标志	OFF : 已完成或未被执行 ON : 执行进行中	● 当低速执行类型程序被执行时, 变为 ON。	S (每个 END 处理)	新增	○
SM551	读取模块服务间隔	OFF : 忽略 ON : 读	● 当此继电器从 OFF 变为 ON 时, 由 SD550 的模块服务间隔被读到从 SD551 到 SD552。	U	新增	○+Rem

(5) 存储卡

特殊继电器列表

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM600	存储卡可用标志	OFF : 不可用 ON : 可用	● 当存储卡 A 准备好由用户使用时为 ON。	S(初始化)	新增	○
SM601	存储卡保护标志	OFF : 无保护 ON : 有保护	● 当存储卡 A 保护变为 ON 时, 变为 ON。	S(初始化)	新增	
SM602	驱动 1 标志	OFF : 无驱动 1 ON : 驱动 1 存在	● 当驱动 1(卡 1RAM 区域)存在时, 变为 ON。	S(初始化)	新增	
SM603	驱动 2 标志	OFF : 无驱动 2 ON : 驱动 2 存在	● 当驱动 2(卡 1ROM 区域)存在时, 变为 ON。	S(初始化)	新增	
SM604	存储卡正被使用标志	OFF : 没有被使用 ON : 处于使用中	● 当存储卡 A 被使用时, 变为 ON。	S (状态改变)	新增	
SM605	存储卡移掉/插入禁止标志	OFF : 移掉/插入允许 ON : 移掉/插入允许	● 当存储卡 A 不能被插入或移掉时, 变为 ON。	U	新增	
SM609	存储卡移掉/插入允许标志	OFF : 移掉/插入禁止 ON : 移掉/插入允许	● 由用户变为 ON 以允许移掉/插入存储卡。 ● 在存储卡被移掉后由系统变为 OFF。 ● 此触点只有在 SM604 和 SM605 为 OFF 时才能使用	U/S	新增	
SM620	驱动 3/4 可用标志	OFF : 不可用 ON : 可用	● 常 ON	S(初始化)	新增	
SM621	驱动 3/4 保护标志	OFF : 无保护 ON : 保护	● 常 OFF	S(初始化)	新增	
SM622	驱动 3 标志	OFF : 无驱动 3 ON : 驱动 3 存在	● 常 ON	S(初始化)	新增	
SM623	驱动 4 标志	OFF : 无驱动 4 ON : 驱动 4 存在	● 常 ON	S(初始化)	新增	QCPU
SM640	文件寄存器使用	OFF : 文件寄存器未被使用 ON : 文件寄存器使用中	● 当文件寄存器处于使用中时, 变为 ON。	S (状态改变)	新增	○
SM650	注释使用	OFF : 文件寄存器未被使用 ON : 文件寄存器使用中	● 当注释文件处于使用中时, 变为 ON。	S (状态改变)	新增	
SM660	引导操作	OFF : 内部存储器执行 ON : 引导操作处理中	● 当引导操作处于处理中时, 变为 ON。 ● 如果引导指定变为 OFF, 则变为 OFF。	S (状态改变)	新增	
SM672	存储卡 A 文件寄存器访问范围标志	OFF : 处于访问范围内 ON : 处于访问范围外	● 当访问处于存储卡 A 的文件寄存器 R 的访问范围区域外时, 变为 ON。 (在 END 处理中设置。) ● 在用户程序中复位。	S/U	新增	

(6) 和特殊继电器相关的指令

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM700	进位标志	OFF : 进位 OFF ON : 进位 ON	● 用于应用指令的进位标志。	S (指令执行)	M9012	○
SM701	输出字符数选择	OFF : 16 字符输出 ON : 输出直到 NUL	● 当 SM701 为 OFF 时, 输出 16 字符的 ASCII 码。 ● 当 SM701 为 ON 时, 执行输出, 直到碰到 ASCII 码的 NUL (00H) 码。	U	M9049	
SM702	搜索方法	OFF : 搜索下一个 ON : 2-部分搜索	● 指定被搜索指令使用的方法。 ● 数据必须按照 2-部分搜索的要求排列。	U	新增	
SM703	排序	OFF : 升序 ON : 降序	● 排序指令用于指定数据是按照升序还是按照降序排列。	U	新增	
SM704	块比较	OFF : 没有找到匹配 ON : 所有的都匹配	● 当 BKCOMP 指令的所有数据条件都满足时, 变为 ON。	S (指令执行)	新增	
SM710	CHK 指令优先级标志	OFF : 条件优先 ON : 模式优先	● 当为 OFF 时保持初始设置。 ● 当为 ON 时进行 CHK 优先级更新。	S (指令执行)	新增	○
SM715	EI 标志	OFF : 处于 DI ON : 处于 EI	● 当 EI 指令被执行时为 ON。	S (指令执行)	新增	○
SM720	注释读完成标志	OFF : 注释读没有被完成 ON : 注释读完成	● 当 COMRD 或 PRC 指令的处理完成时, 变为 ON 一个扫描周期。	S (状态改变)	新增	QCPU
SM721	文件正被访问	OFF : 文件没有被访问 ON : 文件正被访问	● 当一个文件正被 S.FWRITE、S.FREAD、COMRD、PRC 或 LEDC 指令访问时, 变为 ON。	S (状态改变)	新增	
SM722	BIN/DBIN 指令错误禁止标志	OFF : 执行错误检测 ON : 不执行错误检测	● 当“OPERATION ERROR”由于 BIN 或 DBIN 指令而被禁止时, 变为 ON。	U	新增	
SM736	PKEY 指令执行处于处理中的标志	OFF : 指令未被指令 ON : 指令执行	● 当 PKEY 指令正被执行时为 ON。 当 CR 为输入时变为 OFF, 或者当输入字符串达到 32 个字符时。	S (指令执行)	新增	○
SM737	PKEY 指令的键盘输入接收标志	OFF : 使能键盘输入接收 ON : 禁止键盘输入接收	● 当执行键盘输入时, 变为 ON。 当键盘输入已经被存储到 CPU 时, 变为 OFF。	S (指令执行)	新增	
SM775	在 COM 指令执行过程中链接刷新处理的选择	OFF : 执行链接刷新 ON : 无链接刷新被执行	● 选择是只执行用于 COM 指令执行的一般数据处理还是将链接刷新处理一并执行。	U	新增	
SM776	在 CALL 上使能/禁止本地设备	OFF : 本地设备禁止 ON : 本地设备使能	● 决定在 CALL 上被调用程序中的本地设备是否被使能/禁止。	U (状态改变)	新增	
SM777	在中断程序中使能/禁止本地设备	OFF : 本地设备禁止 ON : 本地设备使能	● 决定在中断程序的执行中本地设备是否被使能/禁止。	U (状态改变)	新增	

(7) 调试

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM800	跟踪准备	OFF : 没有准备好 ON : 已准备好	● 当跟踪准备完成时, 变为 ON。	S (状态改变)	新增	QCPU
SM801	跟踪开始	OFF : 挂起 ON : 开始	● 当此继电器变为 ON 时跟踪开始。 ● 当此继电器变为 OFF 时跟踪挂起。 (所有相关的特殊 Ms 变为 OFF。)	U	M9047	QCPU
SM802	跟踪执行进行中	OFF : 挂起 ON : 开始	● 在跟踪执行过程中变为 ON。	S (状态改变)	M9046	QCPU
SM803	跟踪触发器	OFF → ON: 开始	● 当此继电器从 OFF 变为 ON 时跟踪被触发。(和跟踪指令执行状态相同)	U	M9044	QCPU
SM804	跟踪触发器之后	OFF : 不在触发器之后 ON : 在触发器之后	● 在跟踪被触发之后变为 ON。	S (状态改变)	新增	QCPU
SM805	跟踪完成	OFF : 未完成 ON : 结束	● 在跟踪完成时变为 ON。	S (状态改变)	9043	QCPU
SM820	步跟踪准备	OFF : 没有准备好 ON : 已准备好	● 在程序跟踪注册准备好后变为 ON。	U	新增	○
SM826	跟踪错误	OFF : 正常 ON : 错误	● 如果在跟踪执行过程中发生错误则变为 ON。	S (状态改变)	新增	

(8) 闭锁区域

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM910	RKEY 注册标志	OFF : 键盘输入没有被注册 ON : 键盘输入被注册	● 在键盘输入被注册时变为 ON。 如果键盘输入没有被注册则为 OFF。	S (指令执行)	新增	QnA

(9) A 到 Q/QnA 转换的对应

特殊继电器 SM1000 至 SM1255 是在 A 至 Q/QnA 转换后和 ACPU 的特殊继电器 M9000 至 M9255 对应的继电器。

所有这些继电器都由系统控制，这样用户就不能在程序中改变它们的 ON/OFF。

如果用户希望改变这些继电器的 ON/OFF，那么就要修改程序以使用 QCPU/QnACPU 特殊继电器。

可是，对于 SM1084 和 SM1200 至 SM1255，如果用户可以在转换前将某些特殊继电器 M9084 和 M9200 至 M9255 变为 ON/OFF，那么用户也能在转换后将 SM1084 和 SM1200 至 SM1255 中的相应继电器变为 ON/OFF。

对于 ACPU 中的特殊继电器的详细信息，参见每个 CPU 的用户手册，和 MELSECNET 或 MELSECNET /B 数据链接系统参考手册。

要点

当修改后的特殊继电器被用于 QCPU 时，处理时间可能会长一些。当不使用修改后的特殊继电器时，GX Developer 参数中的 PC 系统设置中的“A-系列 CPU 兼容性设置”不被检查。

备注

以下是对修改栏中关于特殊继电器的新增加的解释。

- ① 当一个特殊继电器被用于修改时，软元件号应改为提供的 QCPU/QnACPU 特殊继电器。
- ② 当 ☐ 被提供时，修改后的特殊继电器可以被用于此软元件号。
- ③ 当 ☒ 被提供时，此软元件号不能用于 QCPU/QnACPU。

特殊继电器列表

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义	详细信息	可用的 CPU
M9000	SM1000	—	保险丝烧坏	OFF : 正常 ON : 模块有保险丝被烧坏	<ul style="list-style-type: none">● 当有一个或多个输出模块的保险丝被烧坏时，变为 ON。● 如果此后条件恢复为正常，则保持 ON。● 远程 I/O 站点的输出模块也进行保险丝状况检查。	○
M9002	SM1002	—	I/O 模块验证错误	OFF : 正常 ON : 有错误	<ul style="list-style-type: none">● 如果 I/O 模块的状态和系统上电时的状态不同，则变为 ON。● 如果此后条件恢复为正常，则保持 ON。● 对远程 I/O 站点的模块也进行 I/O 模块确认。● 复位只在特殊寄存器 SD1116 到 SD1123 被复位时才被允许。	
M9004	SM1004	—	NIMI 链接错误	OFF : 正常 ON : 有错误	<ul style="list-style-type: none">● 即使只有一个被安装的 AJ71PT32(S3) 模块被发现有 MINI(S3) 链接错误，也变为 ON。● 如果此后条件恢复为正常，则保持 ON。	QnA
M9005	SM1005	—	AC DOWN 检测	OFF : AC DOWN 没有被检测 ON : AC DOWN 被检测	<ul style="list-style-type: none">● 在使用 AC 电源供电模块时，发生 20ms 内的瞬时电源故障时，变为 ON。● 当电源变为 OFF 时复位，然后为 ON。	○
					<ul style="list-style-type: none">● 在使用 DC 电源供电模块时，发生 10ms 内的瞬时电源故障时，变为 ON。● 当电源变为 OFF 时复位，然后为 ON。	QCPU
					<ul style="list-style-type: none">● 在使用 DC 电源供电模块时，发生 1ms 内的瞬时电源故障时，变为 ON。● 当电源变为 OFF 时复位，然后为 ON。	QnA

特殊继电器列表(续表)

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义	详细信息	可用的 CPU
M9006	SM1006	—	电池电压低	OFF : 正常 ON : 电池电压低	● 当电池电压下降或低于指定值时, 变为 ON。 ● 当电池电压从那以后恢复到正常时, 变为 OFF。	○
M9007	SM1007	—	电池电压低闭锁	OFF : 正常 ON : 电池电压低	● 当电池电压下降或低于指定值时, 变为 ON。 ● 如果电池电压从那以后恢复到正常, 则保持 ON。	
M9008	SM1008	SM1	自诊断错误	OFF : 无错误 ON : 有错误	● 当有错误被当作自诊断结果被发现时, 变为 ON。	
M9009	SM1009	SM62	报警器检测	OFF : 无 F 号码被检测到 ON : F 号码被检测到	● 当 [SET] F 指令的 [OUT] F 被执行时, 变为 ON。 ● 当 SD1124 数据被清零时, 变为 OFF。	
M9011	SM1011	SM56	操作错误标志	OFF : 无错误 ON : 有错误	● 当在应用指令的执行过程中发生操作错误时, 变为 ON。 ● 如果从那以后条件恢复到正常, 则保持 ON。	
M9012	SM1012	SM700	进位标志	OFF : 进位 OFF ON : 进位 ON	● 进位标志用在应用指令中。	
M9016	SM1016		数据存储区清除标志	OFF : 忽略 ON : 输出清除	● 当 SM1016 为 ON 时, 从计算机上清除包括处于远程运行模式的闭锁区域(除特殊继电器和特殊寄存器)内的数据存储区。	
M9017	SM1017		数据存储区清除标志	OFF : 忽略 ON : 输出清除	● 当 SM1016 为 ON 时, 从计算机上清除包括处于远程运行模式的没有被闭锁的区域(除特殊继电器和特殊寄存器)内的数据存储区。	
M9020	SM1020	—	0 号用户定时时钟		● 继电器按照预先设定的扫描间隔重复 ON/OFF。 ● 当电源变为 ON 或执行复位操作时, 时钟从 OFF 开始启动。 ● 由 [DUTY] 指令设定 ON/OFF 的间隔。 <p>n1: ON 的扫描间隔 n2: OFF 的扫描间隔</p>	
M9021	SM1021	—	1 号用户定时时钟			
M9022	SM1022	—	2 号用户定时时钟			
M9023	SM1023	—	3 号用户定时时钟			
M9024	SM1024	—	4 号用户定时时钟			
M9025	SM1025	—	时钟数据设置请求	OFF : 忽略 ON : 设置请求当前被使用	● 在 SM1025 从 OFF 变为 ON 的扫描周期内, 在 [END] 指令被执行后, 将从 SD1025 到 SD1028 的时钟数据写到时钟元件内。	
M9026	SM1026	—	时钟数据错误	OFF : 无错误 ON : 有错误	● 被时钟数据(SD1025 到 SD1028)错误变为 ON。	
M9027	SM1027	—	时钟数据显示	OFF : 忽略 ON : 显示	● 时钟数据被从 SD1025 到 SD1028 内读出, 并且月, 日, 小时, 分钟和秒被显示在 CPU 模块前面的 LED 显示器上。	Q3A Q4A Q4AR
M9028	SM1028	—	时钟数据读请求	OFF : 忽略 ON : 读请求	● 当 SD1028 为 ON 时, 以 BCD 码的形式将时钟数据读入 SD1025 到 SD1028 内。	○
M9029	SM1029		数据通信请求的批处理	OFF : 批处理没有被执行 ON : 批处理被执行	● 在此扫描周期的 END 处理内, SM1029 继电器变为 ON, 使用顺序程序来处理在此扫描周期内接收到的所有数据通信请求。 ● 在运行过程中, 数据通信请求的批处理可以被变为 ON 和 OFF。 ● 默认为 OFF(按照数据通信请求被接收到的次序, 每个 END 处理一次处理一个通信请求。	

特殊继电器列表 (续表)

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义	详细信息	可用的 CPU
M9030	SM1030	—	0.1 秒时钟		<ul style="list-style-type: none">0.1 秒、0.2 秒、1 秒和 2 秒时钟被创建。此继电器并不是每个扫描周期都变为 ON 或 OFF，但是，如果相应的时间已用完，即使是在扫描周期内也会变为 ON 和 OFF。从 OFF 开始启动，当 PLC 电源变为 ON 或 CPU 模块被执行复位。	○
M9031	SM1031	—	0.2 秒时钟			
M9032	SM1032	—	1 秒时钟			
M9033	SM1033	—	2 秒时钟			
M9034	SM1034	—	2n 分钟时钟 (1 分钟时钟) *		<ul style="list-style-type: none">根据在 SD414 中设定的秒数 (默认: n = 30) 在 ON 和 OFF 之间交替切换。并不是在每个扫描周期变为 ON 或 OFF，但是，即使是在扫描周期中，如果相应的时间已经用完，也会变为 ON 或 OFF。当 PLC 供电电源变为 ON 或 CPU 模块执行复位操作时，从 OFF 启动。	
M9036	SM1036	—	常 ON		<ul style="list-style-type: none">在顺序程序中用作初始化和应用指令的空触点。SM1038 和 SM1037 变为 ON 和 OFF，而不管 CPU 模块前部钥匙开关的位置。SM1038 和 SM1039 处于和钥匙开关处于 STOP 位置以外的 RUN 状态一样的条件，并变为 OFF 和 ON。如果钥匙开关位于 STOP 位置切换到 OFF。如果钥匙开关不是位于 STOP 位置，则 SM1038 只 ON 一个扫描周期，SM1039 只 OFF 一个扫描周期。	
M9037	SM1037	—	常 OFF			
M9038	SM1038	—	在 RUN 后，只 ON 一个扫描周期			
M9039	SM1039	—	RUN 标志 (在 RUN 后，只 OFF 一个扫描周期)			
M9040	SM1040	SM206	PAUSE 使能线圈	OFF : PAUSE 无效 ON : PAUSE 使能	<ul style="list-style-type: none">当远程 PAUSE 触点变为 ON 并且 SM204 为 ON 时，PAUSE 模式被设定并且 SM206 被变为 ON。	
M9041	SM1041	SM204	USE 状态触点	OFF : PAUSE 无效 ON : PAUSE 有效		
M9042	SM1042	SM203	STOP 状态触点	OFF : STOP 无效 ON : STOP 有效	<ul style="list-style-type: none">当 RUN 钥匙开关或 RUN/STOP 开关位于 STOP 位置时，变为 ON。	
M9043	SM1043	SM805	STOP 状态触点	OFF : 采样跟踪处理中 ON : 采样跟踪完成	<ul style="list-style-type: none">在 STRA 指令执行后，当采样跟踪执行完由参数预先指定的次数时，变为 ON。 当 STRAR 指令被执行时复位。	
M9044	SM1044	SM803	采样跟踪	OFF → ON STRA 和执行一样 ON → OFF STRAR 和执行一样	<ul style="list-style-type: none">切换到 ON/OFF，SM803 能够执行 STRA / STRAR 指令 (SM803 被外部设备强制变为 ON/OFF) 当从 OFF 切换成 ON 时: STRA 指令 当从 ON 切换成 OFF 时: STRAR 指令 存储在 SD1044 中的值被用作采样跟踪的条件。 在扫描时，在时间 → 时间 (10 ms 单位)	
M9045	SM1045		看门狗定时器 (WDT) 复位	OFF : 不复位 WDT ON : 复位 WDT	<ul style="list-style-type: none">当 ZCOM 指令和数据通信请求批处理 (在扫描周期超过 200 ms 时被使用) 被执行时，SM1015 继电器被变为 ON 以复位 WDT。	
M9046	SM1046	SM802	采样跟踪	OFF : 跟踪未被执行 ON : 跟踪进行中	<ul style="list-style-type: none">在采样跟踪过程中被变为 ON。	
M9047	SM1047	SM801	采样跟踪准备	OFF : 采样跟踪挂起 ON : 采样跟踪启动	<ul style="list-style-type: none">采样跟踪未被执行，除非 SM801 变为 ON。采样跟踪被挂起，当 SM801 变为 OFF 时。	
M9049	SM1049	SM701	输出字符数的选择	OFF : 输出，直到碰到 NULL 代码 ON : 16 字符输出	<ul style="list-style-type: none">当 SM701 为 OFF 时，输出字符直到 NULL (00H) 代码。当 SM701 为 ON 时，输出 16 位字符的 ASCII 码。	
M9051	SM1051		CHG 指令执行禁止	OFF : 使能 ON : 无效	<ul style="list-style-type: none">变为 ON 使 CHG 指令无效。当程序传送被请求使变为 ON。当传送结束时自动变为 OFF。	
M9052	SM1052		SEG 指令切换	OFF : 7SEG 部分显示 ON : I/O 局部刷新	<ul style="list-style-type: none">当 SM1052 为 ON 时，SEG 指令被作为 I/O 局部刷新指令执行。当 SM1052 为 ON 时，SEG 指令被作为 I/O 局部刷新指令执行。	

*: 1 分钟时钟表示 ACPU 的特殊继电器 (M9034) 的名称。

特殊继电器列表 (续表)

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义	详细信息	可用的 CPU
M9054	SM1054	SM205	STEP RUN 标志	OFF : STEP RUN 无效 ON : STEP RUN 有效	● 当 RUN 钥匙开关位于 STEP RUN 位置时, 变为 ON。	QnA
M9055	SM1055	SM808	状态闭锁完成标志	OFF : 未完成 ON : 完成	● 当状态闭锁完成时变为 ON。由复位指令变为 OFF。	
M9056	SM1056		主程序 P、I 设定请求	OFF : 除 P、I 设定被执行外的情况 ON : P、I 设定被执行	● 在运行过程中, 其它程序(例如, 主程序运行时的子程序)的转换完成后提供 P、I 设定请求。 当 P、I 设置完成后自动变为 OFF。	○
M9057	SM1057		子程序 P、I 设定请求	OFF : 除 P、I 设定被执行外的情况 ON : P、I 设定被执行		
M9058	SM1058		主程序 P、I 设定完成	在 P、I 设定完成时立刻变为 ON	● 在 P、I 设定完成后变为 ON, 然后重新变为 OFF。	
M9059	SM1059		子程序 P、I 设定完称	在 P、I 设定完成时立刻变为 ON		
M9060	SM1060		子程序 2 P、I 设置请求	OFF : P、I 设定正被请求外的其它情况 ON : P、I 设定正被请求	● 在运行过程中, 其它程序(例如, 主程序运行时的子程序)的转换完成后提供 P、I 设定请求。当 P、I 设置完成后自动变为 OFF。	
M9061	SM1061		子程序 3 P、I 设定请求	OFF : P、I 设定正被请求外的其它情况 ON : P、I 设定正被请求	● 在运行过程中, 其它程序(例如, 主程序运行时的子程序)的转换完成后提供 P、I 设定请求。当 P、I 设置完成后自动变为 OFF。	
M9065	SM1065	SM711	分开处理执行检测	OFF : 分开处理没有进行 ON : 处于分开处理中	● 当防水屏幕被分开处理转移到 AD57(S1)/AD58 时变为 ON, 并在分开处理完成时变为 OFF。	QnA
M9066	SM1066	SM712	分开处理请求标志	OFF : 批处理 ON : 分开处理	● 当防水屏幕被分开处理转移到 AD57(S1)/AD58 时变为 ON。	
M9070	SM1070		A8UPU/A8PUJ 需要的搜索时间	OFF : 读时间没有缩短 ON : 读时间缩短	● 变为 ON 以缩短在 A8UPU/A8PUJ 中的搜索时间。 (在这种情况下, 扫描时间增加 10%。) * A8UPU/A8PUJ 不能用在 QCPU/QnACPU 特殊继电器中。	○
M9081	SM1081	SM714	通信请求注册区域 BUSY 信号	OFF : 通信请求注册区域中有空闲空间 ON : 通信请求注册区域无空闲空间	● 连接到 AJ71PT32-S3、A2C 或 A52G 的远程终端模块的通信使能/禁止指示。	QnA
M9084	SM1084		错误检查	OFF : 执行错误检查 ON : 无错误检查	● 当 END 指令被处理时, 此继电器被用于设置是否执行错误检查(用于设置 END 指令处理时间)。 ● 保险丝破损检测 ● 电池检查 ● I/O 模块校对检查	○
M9091	SM1091		指令错误标志	OFF : 无错误 ON : 有错误	● 操作错误的详细因素被存储到 SD1091 时, 变为 ON。 ● 如果从那以后条件恢复为正常, 则保持 ON。	
M9094	SM1094	SM251	I/O 模块改变标志	OFF : 替换 ON : 无替换	● 在被改变的 I/O 模块的头 I/O 号码设置到 SD251 后, 当 SM251 变为 ON 时, 这些 I/O 模块可以在线改变。 (在每个设置中只有一个模块可以被改变) ● 当在 RUN 过程中改变 I/O 模块时, 在程序中或在 GX Developer 的测试模式中将此继电器变为 ON。当处于 STOP 时, 在 GX Developer 的测试模式中将此继电器变为 ON。 ● RUN/STOP 模式在 I/O 模块改变完成前不能被改变。	QnA
M9100	SM1100	SM320	存在/不存在 SFC 程序	OFF : SFC 程序未被使用 ON : SFC 程序被使用	● 如果 SFC 程序已登记则变为 ON, 并且在 SFC 未登记时变为 OFF。	○
M9101	SM1101	SM321	启动/停止 SFC 程序	OFF : SFC 程序停止 ON : SFC 程序启动	● 如果 SFC 程序将被启动, 应该由程序将其变为 ON。如果变为 OFF, 则执行步的操作输出变为 OFF 并且 SFC 程序停止。	

特殊继电器列表 (续表)

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义	详细信息	可用的 CPU
M9102	SM1102	SM322	SFC 程序启动状态	OFF : 初始化启动 ON : 继续	<ul style="list-style-type: none">当使用 SM322 重新启动 SFC 程序时, 选择启动步。 ON: 当 SFC 程序停止时所有执行状态都被清除, 并且程序从块 0 的初始步处重新启动。 OFF: 当程序停止时, 从正被执行块的程序步开始。一旦变为 ON, 程序被锁在系统中, 并且即使系统断电也保持为 ON。 系统上电后或者从块 0 的初始步启动时, 应该由顺序程序变为 OFF。	○
M9103	SM1103	SM323	存在/不存在连续转换	OFF : 连续转换无效 ON : 连续转换有效	<ul style="list-style-type: none">当所有连续步的转换条件都建立时, 对转换条件已经建立的步, 选择是连续转换还是一步一步的转换。 ON: 执行连续转换。 OFF: 每个扫描周期转换一步。	
M9104	SM1104	SM324	连续转换挂起标志	OFF : 当转换完成时 ON : 当无转换时	<ul style="list-style-type: none">在操作处于连续转换模式或连续转换时为 OFF, 当没有执行连续转换时为 ON。在没有连续转换模式的操作中常 ON。	
M9108	SM1108	SM90	步转换看门狗定时器启动(相当于 D9108)	OFF : 看门狗定时器复位 ON : 看门狗定时器复位启动	<ul style="list-style-type: none">当步转换看门狗定时器的测量被启动时, 变为 ON。 将此继电器变为 OFF, 以复位步转换看门狗定时器。	
M9109	SM1109	SM91	步转换看门狗定时器启动(相当于 D9109)			
M9110	SM1110	SM92	步转换看门狗定时器启动(相当于 D9110)			
M9111	SM1111	SM93	步转换看门狗定时器启动(相当于 D9111)			
M9112	SM1112	SM94	步转换看门狗定时器启动(相当于 D9112)			
M9113	SM1113	SM95	步转换看门狗定时器启动(相当于 D9113)			
M9114	SM1114	SM96	步转换看门狗定时器启动(相当于 D9114)			
M9180	SM1180	SM825	激活步采样跟踪完成标志	OFF : 跟踪启动 ON : 跟踪完成	<ul style="list-style-type: none">当所有指定块的采样跟踪完成时置位。当采样跟踪启动时复位。	
M9181	SM1181	SM822	激活步采样跟踪执行标志	OFF : 跟踪未被执行 ON : 跟踪正在执行	<ul style="list-style-type: none">当采样跟踪正被执行时置位。 当采样跟踪已完成或挂起时复位。	
M9182	SM1182	SM821	激活步采样跟踪允许	OFF : 跟踪禁止/挂起 ON : 跟踪使能	<ul style="list-style-type: none">选择采样跟踪执行使能/禁止。 ON: 采样跟踪执行被使能。 OFF: 采样跟踪执行被禁止。 如果在采样跟踪执行中变为 OFF, 则跟踪被挂起。	

特殊继电器列表 (续表)

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义			详细信息	可用的 CPU
M9196	SM1196	SM325	在块停止时的操作输出	OFF : 线圈输出 OFF ON : 线圈输出 ON			● 当块停止被执行时选择操作输出。 ON: 通过使用在块停止时被执行步的操作输出, 保持被使用线圈的 ON/OFF 状态。 OFF: 所有线圈输出都变为 OFF。(SET 指令的操作输出的被保持, 而不管 M9196 的 ON/OFF 状态)	○
M9197	SM1197		在烧坏的保险丝和 I/O 校验错误显示之间切换	SM 9197	SM 1198	被显示的 I/O 号码	根据 SM1197 和 SM1198 中 ON/OFF 状态的组合, 在保险丝烧坏模块存储寄存器(SD1100 到 SD1107)中的 I/O 号码和 I/O 模块校验错误存储寄存器(SD1116 到 SD1123)中的 I/O 号码之间切换。	
	OFF			OFF	X/Y 0 到 7F0			
	ON			OFF	X/Y 800 到 FF0			
M9198	SM1198			OFF	ON	X/Y 1000 到 17F0		
				ON	ON	X/Y 1800 到 1FF0		
M9199	SM1199		在线采样跟踪/状态闭锁的数据恢复	OFF : 数据恢复禁止 ON : 数据恢复允许			● 当采样跟踪/状态闭锁被执行时, 在重新启动时恢复 CPU 模块中的设置数据。 ● 为了重新执行, SM1199 应为 ON。(当从外围设备重新写数据时则没有必要)	
M9200	SM1200	—	ZNRD 指令 (LRDP 指令, 用于 ACPU)接收 (用于主站点)	OFF : 不被接收 ON : 被接收			● 取决于 ZNRD (字设备读) 指令是否被接收。 ● 在程序中用作 ZNRD 的互锁指令。 ● 使用 RST 指令进行复位。	QnA
M9201	SM1201	—	ZNRD 指令 (LRDP 指令, 用于 ACPU)完成 (用于主站点)	OFF : 未完成 ON : 结束			● 取决于 ZNRD (字设备读) 指令的执行是否已完成。 ● 在 ZNRD 指令执行完成后, 用作复位 M9200 和 M9201 的条件触点。 ● 使用 RST 指令进行复位。	
M9202	SM1202	—	ZNWR 指令 (LWTP 指令, 用于 ACPU)接收 (用于主站点)	OFF : 不被接收 ON : 被接收			● 取决于 ZNWR (字设备写) 指令是否被接收。 ● 在程序中用作 ZNWR 指令的互锁。 ● 使用 RST 指令进行复位。	
M9203	SM1203	—	ZNWR 指令 (LWTP 指令, 用于 ACPU)完成 (用于主站点)	OFF : 未完成 ON : 结束			● 取决于 ZNWR (字设备写) 指令执行是否已经完成。 ● 用作在 ZNWR 指令完成后复位 M9202 和 M9203 的条件触点。 ● 使用 RST 指令进行复位。	
M9204	SM1204	—	ZNRD 指令 (LRDP 指令, 用于 ACPU)接收 (用于本地站点)	OFF : 未完成 ON : 结束			ON 表明 ZNRD 指令是在本地站点完成的。	
M9205	SM1205	—	ZNWR 指令 (LWTP 指令, 用于 ACPU)	OFF : 未完成 ON : 结束			ON 表明 ZNWR 指令是在本地站点完成的。	
M9206	SM1206	—	自站链接参数错误	OFF : 正常 ON : 不正常			取决于自站的链接参数设置是否有效。	
M9207	SM1207	—	链接参数检查结果	OFF : YES ON : NO			取决于在三级系统中, 第二级主站点的链接参数设置是否和第三级主站点的链接参数设置匹配。 (只对三级系统中的主站点有效。)	
M9208	SM1208	—	设定主站点 B 和 W 的传送范围 (只对低级链接主站点)	OFF : 传送到第 2 级和第 3 级 ON : 只传送到第 2 级			● 取决于由高级一链接主站点 (自站) 控制的 B 和 W 数据是否被送往低级一链接的本地站点 (第三级站点)。 ● 当 SM1208 为 OFF 时 ... 自站的 B 和 W 被送往第三级站点。 ● 当 SM1208 为 ON 时 ... 自站的 B 和 W 不被送往第三级站点。	

特殊继电器列表 (续表)

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义	详细信息	可用的 CPU
M9209	SM1209	—	链接参数检查命令 (只对低级链接主站点)	OFF : 执行检查功能 ON : 检查不执行	<ul style="list-style-type: none"> ● 前向或反向回路测试执行正在进行中。 ● 当 SM1209 为 ON 时, 高级和低级链接的链接参数不被检查。 ● 当 SM1209 为 OFF 时, 高级和低级链接的链接参数被检查。 	QnA
M9210	SM1210	—	链接卡错误 (用于主站点)	OFF : 正常 ON : 不正常	控制是否被执行取决于链接卡硬件是否有故障。	
M9211	SM1211	—	链接模块错误 (用于本地站点)	OFF : 正常 ON : 不正常	控制是否被执行取决于链接卡硬件是否有故障。	
M9224	SM1224	—	链接状态	OFF : 在线 ON : 离线, 站-站测试, 或自-回路返回测试	取决于主站点是在线还是离线, 是处于站-站测试模式还是处于自-回路返回测试模式。	
M9225	SM1225	—	前向回路错误	OFF : 正常 ON : 不正常	取决于前向回路线路的错误条件。	
M9226	SM1226	—	反向回路错误	OFF : 正常 ON : 不正常	取决于反向回路线路的错误条件。	
M9227	SM1227	—	回路测试状态	OFF : 未被执行 ON : 前向或反向回路检测执行正在进行	取决于主站是否正在执行前向或反向回路测试。	
M9232	SM1232	—	本地站点操作状态	OFF : RUN 或 STEP RUN 状态 ON : STOP 或 PAUSE 状态	控制的执行取决于本地站点是否处于 STOP 或 PAUSE 模式。	
M9233	SM1233	—	本地站点错误检测状态	OFF : 无错误 ON : 错误检测	取决于本地站点是否已从其它站点检测到错误。	
M9235	SM1235	—	本地站点, 远程 I/O 站点参数错误检测状态	OFF : 无错误 ON : 错误检测	取决于本地或者远程 I/O 站点是否在主站点中检测到任何链接参数错误。	
M9236	SM1236	—	本地站点, 远程 I/O 站点初始通信状态	OFF : 无通信 ON : 通信进行中	取决于本地或远程 I/O 站点和主站点之间的初始化通信结果。(参数通信, 等)	
M9237	SM1237	—	本地站点, 远程 I/O 站点错误	OFF : 正常 ON : 不正常	取决于本地或远程 I/O 站点的错误条件。	
M9238	SM1238	—	本地站点, 远程 I/O 站点前向或反向回路错误	OFF : 正常 ON : 不正常	取决于本地或远程 I/O 站点前向和反向回路线路的错误条件。	
M9240	SM1240	—	链接状态	OFF : 在线 ON : 离线, 站点-到-站点测试, 或自己-回路返回测试	取决于本地站点是在线 还是离线, 或者是在站点-到-站点测试还是自己-回路返回测试模式	
M9241	SM1241	—	前向回路线路错误	OFF : 正常 ON : 不正常	取决于前向回路线路的错误条件。	
M9242	SM1242	—	反向回路线路错误	OFF : 正常 ON : 不正常	取决于反向回路线路的错误条件。	
M9243	SM1243	—	回路返回执行	OFF : 回路返回未被执行 ON : 回路返回执行	取决于在本地站点是否发生回路返回。	
M9246	SM1246	—	数据未被接收	OFF : 接收 ON : 未接收	取决于从主站点来的数据是否被接收。	
M9247	SM1247	—	数据未被接收	OFF : 接收 ON : 未接收	取决于在三级系统中一个第三级站点是否已从它的主站点接收到数据。	
M9250	SM1250	—	参数未被接收	OFF : 接收 ON : 未接收	取决于是否已经从主站点接收到链接参数。	
M9251	SM1251	—	链接继电器	OFF : 正常 ON : 不正常	取决于本地站点的数据链接条件。	
M9252	SM1252	—	回路测试状态	OFF : 未被执行 ON : 前向或反向回路测试执行正在进行中	取决于本地站点是否正在执行前向或反向回路测试。	

特殊继电器列表 (续表)

ACPU 特殊继电器	转换后的特殊继电器	用于修改的特殊继电器	名称	含义	详细信息	可用的 CPU
M9253	SM1253	—	主站点操作状态	OFF : RUN 或 STEP RUN 状态 ON : STOP 或 PAUSE 状态	控制是否被执行取决于主站点是否处于 STOP 或 PAUSE 模式。	QnA
M9254	SM1254	—	非自站的本地站点操作状态	OFF : RUN 或 STEP RUN 状态 ON : RUN 或 STEP RUN 状态	控制是否被执行取决于非主站点的本地站点是否处于 STOP 或 PAUSE 模式。	
M9255	SM1255	—	非自站的本地站点错误	OFF : 正常 ON : 不正常	控制是否被执行取决于非自站的本地站点是否出错。	

(10) 过程控制指令

号码	名称	含义	解释	设定者 (当被设定时)	ACPU M9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	可用的 CPU
SM1500	保持模式	OFF : 无保持 ON : 保持	● 对 S. IN 指令范围检测发生范围溢出时, 指定是否保持输出值。	U	新增	○
SM1501	保持模式	OFF : 无保持 ON : 保持	● 对 S. OUT 指令范围检测发生范围溢出时, 指定是否保持输出值。	U	新增	

附录 4 特殊寄存器列表

附录 4.1 基本型 QCPU 的特殊寄存器列表

特殊寄存器，SD，是应用固化在 PLC 中的内部寄存器。
 由于这个原因，它们不能以普通寄存器那样的方式被顺控程序使用。
 然而，数据可以按照需要被写入以控制 CPU 模块。
 如果没有特殊指定，数据以 BIN 值的形式存储在特殊寄存器内。

下表中标题的含义如下所示。

项目	项目的功能
号码	● 表示特殊寄存器的号码
名称	● 表示特殊寄存器的名称
含义	● 表示特殊寄存器的内容
解释	● 讨论特殊寄存器内容的详细信息。
设置者(当被设置时)	<ul style="list-style-type: none"> ● 表示寄存器是被系统设置还是被用户设置，并且，如果是被系统设置，当设置被执行时。 ＜设置者＞ <ul style="list-style-type: none"> S : 由系统设置 U : 由用户设置(从 GX Developer 或类似设备上的顺序程序或者测试操作中) S/U : 由系统和用户设置 ＜当被设置时＞ → 只有当寄存器由系统设置时才被指示 <ul style="list-style-type: none"> 每个 END : 在每条 END 处理过程中被设置 初始化 : 在初始化处理时被设置(当系统电源变为 ON, 或者当从 STOP 变成 RUN 时) 状态改变 : 只有在状态有改变时才被设置 错误 : 只有在错误产生时才被设置 指令执行 : 在指令被执行时被设置 请求 : 只有在有用户请求时才被设置(通过 SM, 等)
相应的 CPU	<ul style="list-style-type: none"> ● 表示相应的 CPU 模块类型名称。 ○ : 可以被应用到所有类型的 CPU 模块 每个 CPU 模块类型名称 : 只能被应用到指定的 CPU 模块

对于下列项目的详细信息，参见下列手册：
 ● 网络 → ● 用于和 Q 系列相应的 MELSECNET/H 网络系统参考手册 (PLC 至 PLC 网络)

特殊寄存器列表

(1) 诊断信息

号码	名称	含义	解释	设定者 (当被设定时)	相应的 CPU
SD0	诊断错误	诊断错误	<ul style="list-style-type: none">● 诊断发现的错误的错误代码以 BIN 数据形式存储。● 内容和最新的故障历史信息相同。	S (错误)	
SD1	诊断错误发生的 时钟时间	诊断错误发生的时钟 时间	<ul style="list-style-type: none">● SD0 数据被更新时的年(最后两位数)和月以 BCD 2 位码形式存储。 (例子) b15 到 b8 b7 到 b0 年(0到99) 月(1到12) :十月, 1995 H9510	S (错误)	
SD2			<ul style="list-style-type: none">● SD0 数据被更新时的日期和小时以 BCD 2 位码形式存储。 (例子) b15 到 b8 b7 到 b0 日期(1到31) 小时(0到23) :10p. m. 在25号 H2510		
SD3			<ul style="list-style-type: none">● SD0 数据被更新时的分钟和秒以 BCD 2 位码的形式存储。 (例子) b15 到 b8 b7 到 b0 分钟(0到59) 秒(0到59) :35分 48秒 (在几点后又过了) H3548		
SD4	错误信息分类	错误信息分类代码	<ul style="list-style-type: none">● 用于标志存储在公用信息区域(SD5 到 SD15)和单独信息区域(SD16 到 SD26)的错误信息是哪分类型的分类代码存储在这里。 b15 到 b8 b7 到 b0 单独信息种类代码 公用信息种类代码● 公用信息分类代码存储了下列一些代码: 0 : 无错误 1 : 单元/模块号码/PLC 号码/基板号码 2 : 文件名/驱动器名 3 : 时间(设定值) 4 : 程序错误定位● 单个信息分类代码存储了下列一些代码: 0 : 无错误 1 : (打开) 2 : 文件名/驱动器名 3 : 时间(实际测量值) 4 : 程序错误定位 5 : 参数号 6 : 报警器号	S (错误)	○

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定时)	相应的 CPU																																																																																																					
SD5	错误公用信息	错误公用信息	<ul style="list-style-type: none">● 和错误代码(SD0)相应的公用信息存储在这里。● 接下来的四种类型的信息存储在这里： <p>① 插槽号</p> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD5</td><td>插槽号/基板号码*1*2</td></tr><tr><td>SD6</td><td>I/O号*3</td></tr><tr><td>SD7</td><td rowspan="9">(空白)</td></tr><tr><td>SD8</td></tr><tr><td>SD9</td></tr><tr><td>SD10</td></tr><tr><td>SD11</td></tr><tr><td>SD12</td></tr><tr><td>SD13</td></tr><tr><td>SD14</td></tr><tr><td>SD15</td></tr></table> <p>(不用作基板号码)</p> <p>* 1: 在多 CPU 系统中, 插槽号或者 CPU 号依据发生的错误被保存。 多 CPU 系统中的插槽 0 指的是 CPU 模块右边最右端的插槽。(参见用于检查哪个号码被存储的错误代码)第 1 CPU: 1, 第 2 CPU: 2, 第 3 CPU: 3</p> <p>* 2: 当 255 被存储进 SD5 时, 表示有指令被执行, 此指令被用于模块能被安装的最后插槽之后的模块。</p> <p>* 3: 当 0FFFFh 被存储进 SD6(I/O 号码), 此 I/O 号码并不是依据 I/O 号码重叠或像 I/O 分配参数那样被鉴别。使用 SD5 来验证错误位置。</p> <p>② 文件名/驱动器名</p> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD5</td><td>驱动器</td></tr><tr><td>SD6</td><td rowspan="4">文件名 (ASCII码: 8字符)</td></tr><tr><td>SD7</td></tr><tr><td>SD8</td></tr><tr><td>SD9</td></tr><tr><td>SD10</td><td>扩展*4</td></tr><tr><td>SD11</td><td>2EH(.) (ASCII 码: 3字符)</td></tr><tr><td>SD12</td><td rowspan="4">(空白)</td></tr><tr><td>SD13</td></tr><tr><td>SD14</td></tr><tr><td>SD15</td></tr></table> <p>(例子)</p> <p>文件名= MAIN, QPG b15 到 b8 b7 到 b0</p> <table><tr><td>41H(A)</td><td>4DH(M)</td></tr><tr><td>4EH(N)</td><td>49H(I)</td></tr><tr><td>20H(SP)</td><td>20H(SP)</td></tr><tr><td>20H(SP)</td><td>20H(SP)</td></tr><tr><td>51H(Q)</td><td>2EH(.)</td></tr><tr><td>47H(G)</td><td>50H(P)</td></tr></table> <p>③ 时间 (设定值)</p> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD5</td><td>时间: 1μs 单位 (0到999μs)</td></tr><tr><td>SD6</td><td>时间: 1ms单位 (0到65535 ms)</td></tr><tr><td>SD7</td><td rowspan="9">(空白)</td></tr><tr><td>SD8</td></tr><tr><td>SD9</td></tr><tr><td>SD10</td></tr><tr><td>SD11</td></tr><tr><td>SD12</td></tr><tr><td>SD13</td></tr><tr><td>SD14</td></tr><tr><td>SD15</td></tr></table> <p>④ 程序错误定位</p> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD5</td><td rowspan="4">文件名 (ASCII码: 8字符)</td></tr><tr><td>SD6</td></tr><tr><td>SD7</td></tr><tr><td>SD8</td></tr><tr><td>SD9</td><td>扩展*4</td></tr><tr><td>SD10</td><td>2EH(.) (ASCII码: 3字符)</td></tr><tr><td>SD11</td><td>模式*5</td></tr><tr><td>SD12</td><td>块号码</td></tr><tr><td>SD13</td><td>步号码/转换号码</td></tr><tr><td>SD14</td><td>顺序步号码(L)</td></tr><tr><td>SD15</td><td>顺序步号码(H)</td></tr></table> <p>* 5: 模式数据</p> <table><tr><td>15</td><td>14</td><td>to</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>← (位号码)</td></tr><tr><td>0</td><td>0</td><td>to</td><td>0</td><td>0</td><td>*</td><td>*</td><td>*</td><td>*</td></tr></table> <p>(未被使用)</p> <p>— SFC块指定 (1)/未指定 (0) — SFC步指定 (1)/未指定 (0) — SFC转换指定 (1)/未指定 (0)</p>	号码	含义	SD5	插槽号/基板号码*1*2	SD6	I/O号*3	SD7	(空白)	SD8	SD9	SD10	SD11	SD12	SD13	SD14	SD15	号码	含义	SD5	驱动器	SD6	文件名 (ASCII码: 8字符)	SD7	SD8	SD9	SD10	扩展*4	SD11	2EH(.) (ASCII 码: 3字符)	SD12	(空白)	SD13	SD14	SD15	41H(A)	4DH(M)	4EH(N)	49H(I)	20H(SP)	20H(SP)	20H(SP)	20H(SP)	51H(Q)	2EH(.)	47H(G)	50H(P)	号码	含义	SD5	时间: 1μs 单位 (0到999μs)	SD6	时间: 1ms单位 (0到65535 ms)	SD7	(空白)	SD8	SD9	SD10	SD11	SD12	SD13	SD14	SD15	号码	含义	SD5	文件名 (ASCII码: 8字符)	SD6	SD7	SD8	SD9	扩展*4	SD10	2EH(.) (ASCII码: 3字符)	SD11	模式*5	SD12	块号码	SD13	步号码/转换号码	SD14	顺序步号码(L)	SD15	顺序步号码(H)	15	14	to	4	3	2	1	0	← (位号码)	0	0	to	0	0	*	*	*	*	S(错误)	○
号码			含义																																																																																																							
SD5			插槽号/基板号码*1*2																																																																																																							
SD6			I/O号*3																																																																																																							
SD7			(空白)																																																																																																							
SD8																																																																																																										
SD9																																																																																																										
SD10																																																																																																										
SD11																																																																																																										
SD12																																																																																																										
SD13																																																																																																										
SD14																																																																																																										
SD15																																																																																																										
号码			含义																																																																																																							
SD5			驱动器																																																																																																							
SD6	文件名 (ASCII码: 8字符)																																																																																																									
SD7																																																																																																										
SD8																																																																																																										
SD9																																																																																																										
SD10	扩展*4																																																																																																									
SD11	2EH(.) (ASCII 码: 3字符)																																																																																																									
SD12	(空白)																																																																																																									
SD13																																																																																																										
SD14																																																																																																										
SD15																																																																																																										
41H(A)	4DH(M)																																																																																																									
4EH(N)	49H(I)																																																																																																									
20H(SP)	20H(SP)																																																																																																									
20H(SP)	20H(SP)																																																																																																									
51H(Q)	2EH(.)																																																																																																									
47H(G)	50H(P)																																																																																																									
号码	含义																																																																																																									
SD5	时间: 1μs 单位 (0到999μs)																																																																																																									
SD6	时间: 1ms单位 (0到65535 ms)																																																																																																									
SD7	(空白)																																																																																																									
SD8																																																																																																										
SD9																																																																																																										
SD10																																																																																																										
SD11																																																																																																										
SD12																																																																																																										
SD13																																																																																																										
SD14																																																																																																										
SD15																																																																																																										
号码	含义																																																																																																									
SD5	文件名 (ASCII码: 8字符)																																																																																																									
SD6																																																																																																										
SD7																																																																																																										
SD8																																																																																																										
SD9	扩展*4																																																																																																									
SD10	2EH(.) (ASCII码: 3字符)																																																																																																									
SD11	模式*5																																																																																																									
SD12	块号码																																																																																																									
SD13	步号码/转换号码																																																																																																									
SD14	顺序步号码(L)																																																																																																									
SD15	顺序步号码(H)																																																																																																									
15	14	to	4	3	2	1	0	← (位号码)																																																																																																		
0	0	to	0	0	*	*	*	*																																																																																																		

* 4: 对于扩展的, 参考下页中的备注。

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定时)	相应的 CPU																																																																																																																																	
SD16	错误单独信息	错误单独信息	<div>● 和错误代码(SD0)相应的单独信息存储在此。</div> <div>① 文件名/驱动器名</div> <div>(例子)</div> <div>文件名 = MAIN.QPG b15 到 8b b7 到 0b</div> <table><tr><th>号码</th><th>含义</th><th>41H(A)</th><th>4DH(M)</th></tr><tr><td>SD16</td><td>驱动器</td><td></td><td></td></tr><tr><td>SD17</td><td rowspan="4">文件名 (ASCII码: 8字符)</td><td>4EH(N)</td><td>49H(I)</td></tr><tr><td>SD18</td><td>20H(SP)</td><td>20H(SP)</td></tr><tr><td>SD19</td><td>20H(SP)</td><td>20H(SP)</td></tr><tr><td>SD20</td><td>20H(SP)</td><td>20H(SP)</td></tr><tr><td>SD21</td><td>扩展*4</td><td>2EH(.)</td><td>51H(Q)</td></tr><tr><td>SD22</td><td>(ASCII码: 3字符)</td><td></td><td>2EH(.)</td></tr><tr><td>SD23</td><td rowspan="4">(空白)</td><td>47H(G)</td><td>50H(P)</td></tr><tr><td>SD24</td><td></td><td></td></tr><tr><td>SD25</td><td></td><td></td></tr><tr><td>SD26</td><td></td><td></td></tr></table> <div>② 时间(实际测量值)</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td>时间: 1μs单位(0到999μs)</td></tr><tr><td>SD17</td><td>时间: 1ms单元(0到65535ms)</td></tr><tr><td>SD18</td><td rowspan="9">(空白)</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td></tr><tr><td>SD21</td></tr><tr><td>SD22</td></tr><tr><td>SD23</td></tr><tr><td>SD24</td></tr><tr><td>SD25</td></tr><tr><td>SD26</td></tr></table> <div>③ 程序错误定位</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td rowspan="4">文件名 (ASCII码: 8字符)</td></tr><tr><td>SD17</td></tr><tr><td>SD18</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td><td>扩展*4</td></tr><tr><td>SD21</td><td>(ASCII码: 3字符)</td></tr><tr><td>SD22</td><td>模式*5</td></tr><tr><td>SD23</td><td>块号码</td></tr><tr><td>SD24</td><td>步号码/转换号码</td></tr><tr><td>SD25</td><td>顺序步号码(L)</td></tr><tr><td>SD26</td><td>顺序步号码(H)</td></tr></table> <div>*5: 模式数据</div> <table><tr><td>15 14 to 4 3 2 1 0</td><td>← (位号码)</td></tr><tr><td>0 0 to 0 0 * * *</td><td></td></tr></table> <div>(未被使用)</div> <div>SFC块指定(1)/未指定(0)</div> <div>SFC步指定(1)/未指定(0)</div> <div>SFC转换指定(1)/未指定(0)</div> <div>④ 参数号码</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td>参数号码*6</td></tr><tr><td>SD17</td><td rowspan="10">(空白)</td></tr><tr><td>SD18</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td></tr><tr><td>SD21</td></tr><tr><td>SD22</td></tr><tr><td>SD23</td></tr><tr><td>SD24</td></tr><tr><td>SD25</td></tr><tr><td>SD26</td></tr></table> <div>⑤ 报警器号码</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td>号码</td></tr><tr><td>SD17</td><td rowspan="10">(空白)</td></tr><tr><td>SD18</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td></tr><tr><td>SD21</td></tr><tr><td>SD22</td></tr><tr><td>SD23</td></tr><tr><td>SD24</td></tr><tr><td>SD25</td></tr><tr><td>SD26</td></tr></table> <div>⑥ 智能功能模块参数错误</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td>参数号码*6</td></tr><tr><td>SD17</td><td>智能功能模块的错误代码</td></tr><tr><td>SD18</td><td rowspan="9">(空白)</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td></tr><tr><td>SD21</td></tr><tr><td>SD22</td></tr><tr><td>SD23</td></tr><tr><td>SD24</td></tr><tr><td>SD25</td></tr><tr><td>SD26</td></tr></table> <div>*6 关于参数号码的详细信息, 参加用户使用的 CPU 模块的手册(功能函数介绍, 编程基础)</div>	号码	含义	41H(A)	4DH(M)	SD16	驱动器			SD17	文件名 (ASCII码: 8字符)	4EH(N)	49H(I)	SD18	20H(SP)	20H(SP)	SD19	20H(SP)	20H(SP)	SD20	20H(SP)	20H(SP)	SD21	扩展*4	2EH(.)	51H(Q)	SD22	(ASCII码: 3字符)		2EH(.)	SD23	(空白)	47H(G)	50H(P)	SD24			SD25			SD26			号码	含义	SD16	时间: 1μs单位(0到999μs)	SD17	时间: 1ms单元(0到65535ms)	SD18	(空白)	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26	号码	含义	SD16	文件名 (ASCII码: 8字符)	SD17	SD18	SD19	SD20	扩展*4	SD21	(ASCII码: 3字符)	SD22	模式*5	SD23	块号码	SD24	步号码/转换号码	SD25	顺序步号码(L)	SD26	顺序步号码(H)	15 14 to 4 3 2 1 0	← (位号码)	0 0 to 0 0 * * *		号码	含义	SD16	参数号码*6	SD17	(空白)	SD18	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26	号码	含义	SD16	号码	SD17	(空白)	SD18	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26	号码	含义	SD16	参数号码*6	SD17	智能功能模块的错误代码	SD18	(空白)	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26	S(错误)	○
号码			含义	41H(A)	4DH(M)																																																																																																																																	
SD16			驱动器																																																																																																																																			
SD17			文件名 (ASCII码: 8字符)	4EH(N)	49H(I)																																																																																																																																	
SD18				20H(SP)	20H(SP)																																																																																																																																	
SD19				20H(SP)	20H(SP)																																																																																																																																	
SD20				20H(SP)	20H(SP)																																																																																																																																	
SD21			扩展*4	2EH(.)	51H(Q)																																																																																																																																	
SD22			(ASCII码: 3字符)		2EH(.)																																																																																																																																	
SD23			(空白)	47H(G)	50H(P)																																																																																																																																	
SD24																																																																																																																																						
SD25																																																																																																																																						
SD26																																																																																																																																						
号码			含义																																																																																																																																			
SD16			时间: 1μs单位(0到999μs)																																																																																																																																			
SD17			时间: 1ms单元(0到65535ms)																																																																																																																																			
SD18			(空白)																																																																																																																																			
SD19																																																																																																																																						
SD20																																																																																																																																						
SD21																																																																																																																																						
SD22																																																																																																																																						
SD23																																																																																																																																						
SD24																																																																																																																																						
SD25																																																																																																																																						
SD26																																																																																																																																						
号码			含义																																																																																																																																			
SD16	文件名 (ASCII码: 8字符)																																																																																																																																					
SD17																																																																																																																																						
SD18																																																																																																																																						
SD19																																																																																																																																						
SD20	扩展*4																																																																																																																																					
SD21	(ASCII码: 3字符)																																																																																																																																					
SD22	模式*5																																																																																																																																					
SD23	块号码																																																																																																																																					
SD24	步号码/转换号码																																																																																																																																					
SD25	顺序步号码(L)																																																																																																																																					
SD26	顺序步号码(H)																																																																																																																																					
15 14 to 4 3 2 1 0	← (位号码)																																																																																																																																					
0 0 to 0 0 * * *																																																																																																																																						
号码	含义																																																																																																																																					
SD16	参数号码*6																																																																																																																																					
SD17	(空白)																																																																																																																																					
SD18																																																																																																																																						
SD19																																																																																																																																						
SD20																																																																																																																																						
SD21																																																																																																																																						
SD22																																																																																																																																						
SD23																																																																																																																																						
SD24																																																																																																																																						
SD25																																																																																																																																						
SD26																																																																																																																																						
号码	含义																																																																																																																																					
SD16	号码																																																																																																																																					
SD17	(空白)																																																																																																																																					
SD18																																																																																																																																						
SD19																																																																																																																																						
SD20																																																																																																																																						
SD21																																																																																																																																						
SD22																																																																																																																																						
SD23																																																																																																																																						
SD24																																																																																																																																						
SD25																																																																																																																																						
SD26																																																																																																																																						
号码	含义																																																																																																																																					
SD16	参数号码*6																																																																																																																																					
SD17	智能功能模块的错误代码																																																																																																																																					
SD18	(空白)																																																																																																																																					
SD19																																																																																																																																						
SD20																																																																																																																																						
SD21																																																																																																																																						
SD22																																																																																																																																						
SD23																																																																																																																																						
SD24																																																																																																																																						
SD25																																																																																																																																						
SD26																																																																																																																																						

备注

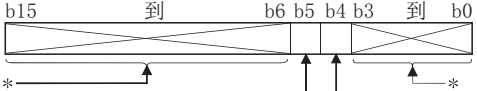
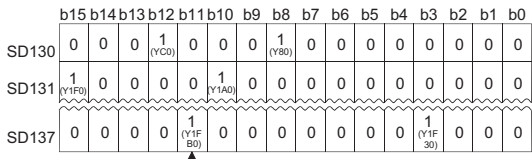
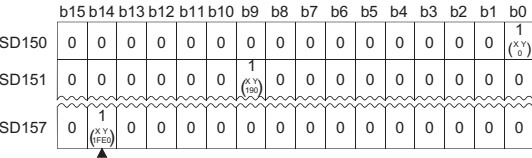
*4 扩展寄存器如下所列

SD10		SD11		扩展名称	文件类型
高 8 位	低 8 位	高 8 位	低 8 位		
51H	50H	41H	40H	QPA	参数
51H	50H	47H	46H	QPG	顺序程序/SFC 程序
51H	43H	44H	42H	QCD	设备注释
51H	44H	52H	50H	QDR	文件寄存器

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定时)	相应的 CPU
SD50	错误复位	执行错误复位的错误 号码	● 存储执行错误复位的错误号码	U	○
SD51	电池电压低闭 锁	位模式表示哪里发 生了电池电压下降	● 当电池电压下降发生时，相应的位变为 1(ON)。 ● 保持为 1，如果电池电压此后恢复为正常。 <div><div>b15</div><div>到</div><div>b0</div><div>0</div><div>CPU错误</div></div>	S(错误)	
SD52	电池电压低	位模式表示哪里发 生了电池电压下降	● 和上边的 SD51 一样的配置 ● 变为 0(OFF)，当电池电压此后恢复为正常。	S(错误)	
SD53	AC/DC DOWN 检 测	AC/DC DOWN 的次 数	● 在 CPU 模块的计算过程中，每当输入电压下降到或低于额定值的 85%(AC 电 源)/65%(DC 电源)，此值便增加一个并存储进 BIN。	S(错误)	
SD60	烧坏的保险丝 号码	保险丝烧坏的模块的 号码	● 存储在这里的值是有烧坏保险丝的模块的最低站点 I/O 号码。	S(错误)	
SD61	I/O 模块验证错 误号码	I/O 模块验证错误模 块号码	● I/O 模块验证发生处的模块 (F 号码) 的最低 I/O 号码。	S(错误)	
SD62	报警器号码	报警器号码	● 检测到第一个报警器的号码被存储在这里。	S (指令执行)	
SD63	报警器的数量	报警器的数量	● 存储搜索到的报警器的数量。		
SD64	检测到的报警 器号码列表	报警器检测号码	当 F 基于 OUT F 或 SET F 变为 ON 时，从 SD64 到 SD79 连续变为 ON 的 F 号 码被登记。 由 RST F 变为 OFF 的 F 号码被从 SD64 SD79 中删除，并且在被删除的 F 号码 被移位到前面的寄存器后此 F 号码被存储。 当 16 个报警器被检测到时，第 17 个检测不再保存到从 SD64 到 SD79 中。 <div><div>SET SET SET RST SET SET SET SET SET SET SET RST</div><div>F50 F25 F99 F25 F15 F70 F65 F38 F110F151F210F50</div><div><div>SD62</div><div>0</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>99</div><div>...</div><div>(检测到 的号码)</div></div><div><div>SD63</div><div>0</div><div>1</div><div>2</div><div>3</div><div>2</div><div>3</div><div>4</div><div>5</div><div>6</div><div>7</div><div>8</div><div>9</div><div>8</div><div>...</div><div>(检测到的报 警器号码)</div></div><div><div>SD64</div><div>0</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>99</div><div data-kind="parent" data-rs="14">(检测到的号码)</div></div><div><div>SD65</div><div>0</div><div>0</div><div>25</div><div>25</div><div>99</div><div>99</div><div>99</div><div>99</div><div>99</div><div>99</div><div>99</div><div>99</div><div>15</div><div data-kind="ghost"></div></div><div><div>SD66</div><div>0</div><div>0</div><div>0</div><div>99</div><div>0</div><div>15</div><div>15</div><div>15</div><div>15</div><div>15</div><div>15</div><div>15</div><div>70</div><div data-kind="ghost"></div></div><div><div>SD67</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>70</div><div>70</div><div>70</div><div>70</div><div>70</div><div>70</div><div>65</div><div data-kind="ghost"></div></div><div><div>SD68</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>65</div><div>65</div><div>65</div><div>65</div><div>65</div><div>65</div><div>38</div><div data-kind="ghost"></div></div><div><div>SD69</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>38</div><div>38</div><div>38</div><div>38</div><div>38</div><div>110</div><div data-kind="ghost"></div></div><div><div>SD70</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>110</div><div>110</div><div>110</div><div>151</div><div data-kind="ghost"></div></div><div><div>SD71</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>151</div><div>151</div><div>210</div><div data-kind="ghost"></div></div><div><div>SD72</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>210</div><div>0</div><div data-kind="ghost"></div></div><div><div>SD73</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div data-kind="ghost"></div></div><div><div>SD74</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div data-kind="ghost"></div></div><div><div>SD75</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div data-kind="ghost"></div></div><div><div>SD76</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div data-kind="ghost"></div></div><div><div>SD77</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div data-kind="ghost"></div></div><div><div>SD78</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div data-kind="ghost"></div></div><div><div>SD79</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div data-kind="ghost"></div></div></div>		
SD65					
SD66					
SD67					
SD68					
SD69					
SD70					
SD71					
SD72					
SD73					
SD74					
SD75					
SD76					
SD77					
SD78					
SD79					

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定时)	相应的 CPU
SD100	传送速度存储区域	存储在串行通讯设置中指定的传送速度。	K96: 9.6kbps, K192: 19.2kbps, K384: 38.4kbps, K576: 57.6kbps, K1152: 115.2kbps	S(上电或复位时)	○
SD101	通讯设置存储区域	存储在串行通讯设置中指定的通讯设置。	 <p>在线程序修正设置 0: 禁止 1: 使能</p> <p>和数校验是/否 0: 否 1: 是</p> <p>*: 从数据被系统使用开始, 它被重新定义。</p>	S(上电或复位时)	
SD102	消息等待时间存储区域	存储在串行通讯设置中指定的消息等待时间。	0: 无等待时间 1 到 FH: 等待时间(单位: 10ms) 默认值为 0	S(上电或复位时)	
SD110	数据发送结果存储区域	当串行通讯功能被使用时, 存储数据发送结果。	数据传送时的错误代码被保存。	S(错误)	
SD111	数据接收结果存储区域	当串行通讯功能被使用时, 存储数据接收结果。	存储数据接收时的错误代码。	S(错误)	
SD130	保险丝烧坏的模块	16 点数据单元中的位模式, 表示了其保险丝被烧坏的模块。 0: 无烧坏的保险丝 1: 有烧坏的保险丝	<ul style="list-style-type: none"> ● 保险丝被烧坏的输出模块的号码以位模式被输入(在 16 点数据单元中)。(如果模块的号码是由参数设定的, 此参数设定号码被存储。) ● 远程站点输出模块中的保险丝烧坏的情况也被检测。  <p>表示一个烧坏的保险丝</p>	S(错误)	
SD131					
SD132					
SD133					
SD134					
SD135					
SD136			<ul style="list-style-type: none"> ● 即使烧坏的保险丝被新保险丝替换掉, 此标志位也不被清除。此标志由错误复位操作清除。 		
SD137	I/O 模块验证错误	在 16 位数据单元中的位模式, 表示有验证错误的模块。 0: 无 I/O 验证错误 1: I/O 验证错误存在	<ul style="list-style-type: none"> ● 当系统上电时, 其信息和登记的 I/O 模块信息不同的模块号码被设置在此寄存器内(在 16 位数据单元内)。(如果 I/O 号码由参数设定, 此参数设定号码被存储。) ● 同时检测 I/O 模块信息  <p>表示一个 I/O 模块确认错误</p>	S(错误)	
SD150					
SD151					
SD152					
SD153					
SD154					
SD155					
SD156			<ul style="list-style-type: none"> ● 即使烧坏的保险丝被新保险丝替换掉, 此标志位也不被清除。此标志由错误复位操作清除。 		
SD157					

(2) 系统信息

特殊寄存器列表

号码	名称	含义	解释	设定者 (当被设定时)	相应的 CPU
SD200	开关的状态	CPU 开关的状态	<div>● CPU 模块开关状态按照下列格式存储:</div> <div><div>b15 到 b8 b7 到 b4 b3 到 b0</div><div>空白 ② ①</div></div> <div>①: CPU 开关状态 0: RUN 1: STOP</div> <div>②: 存储卡开关 常 OFF</div>	S(每个 END 处理)	
SD201	LED 状态	CPULED 的状态	<div>● 下列位模式用于存储 CPU 模块上 LED 的状态。</div> <div><div>b15 到 b4 b3 到 b0</div><div>空白 ② ①</div></div> <div>①: RUN</div> <div>②: ERR.</div>	S (状态改变)	○
SD203	CPU 的工作状态	CPU 工作状态	<div>● CPU 模块的工作状态按照下列图表中所示的格式存储:</div> <div><div>b15 到 b12 b11 到 b8 b7 到 b4 b3 到 b0</div><div>② ①</div></div> <div>①: CPU 工作状态 0 : RUN 1 : 空白 2 : STOP 3 : PAUSE</div> <div>②: STOP/PAUSE 原因 0 : RUN/STOP/RESET 切换 1 : 远程触点 2 : 来自其它一些远程源的 GX Developer/串行通讯模块。 3 : 内部程序指令</div> <div>注意: 优先级为最早的最先 4 : 错误</div>	S(每个 END 处理)	
SD207	LED 显示优先级	优先级 1 到 4	<div>● 在带有错误项目号码的错误发生时, 设置 LED 显示部分的 ON (闪烁) 优先级。</div> <div>● 只适用报警器(错误项目号码为 7)。优先级设置区域显示如下。</div> <div><div>b15 到 b12 b11 到 b8 b7 到 b4 b3 到 b0</div><div><div>SD207 优先级4 优先级3 优先级2 优先级1</div><div>SD208 优先级8 优先级7 优先级6 优先级5</div><div>SD209 优先级10 优先级9</div></div><div>默认值 SD207 = 0000H SD208 = 0700H SD209 = 0000H</div></div> <div>● 有“7”被设置到从 1 到 10 之间的任一优先级中, 当报警器变为 ON 时 ERR. LED 变为 ON。</div> <div>● 无“7”被设置到从 1 到 10 之间的任一优先级中, 如果报警器变为 ON, ERR. LED 不会变为 ON</div>	U	系列号为 04122 或更大
SD208		优先级 5 到 8			
SD209		优先级 9 到 10			

特殊寄存器列表(续表)

号码	号码	含义	解释	设定者 (当被设定时)	相应的 CPU																								
SD210	时钟数据	时钟数据(年, 月)	<ul style="list-style-type: none">年(后两为数字)和月如下列所示, 被以 BCD 码的形式存储在 SD210 中: <div><div><div>b15 到 b12</div><div>b11 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div>例子: July 1993 9307_H</div></div>	S/U (请求)																									
SD211	时钟数据	时钟数据(日期, 小时)	<ul style="list-style-type: none">日期和小时如下列所示, 被以 BCD 码的形式存储在 SD211 中: <div><div><div>b15 到 b12</div><div>b11 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div>例子: 31号, 10下午 3110_H</div></div>																										
SD212	时钟数据	时钟数据(分钟, 秒)	<ul style="list-style-type: none">分钟和秒(显示的小时数后的)如下列所示, 被以 BCD 码的形式存储在 SD212 中: <div><div><div>b15 到 b12</div><div>b11 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div>例子: 35分, 48秒 3548_H</div></div>																										
SD213	时钟数据	时钟数据 (星期几)	<ul style="list-style-type: none">年(两位数)和星期几如下所示, 被以 BCD 码的形式存储在 SD213 中: <div><div><div>b15 到 b12</div><div>b11 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div>例子: 1993, 星期五 1905_H</div><div>年的高位数 (0到99)</div><table><tr><th>星期几</th></tr><tr><td>0 星期日</td></tr><tr><td>1 星期一</td></tr><tr><td>2 星期二</td></tr><tr><td>3 星期三</td></tr><tr><td>4 星期四</td></tr><tr><td>5 星期五</td></tr><tr><td>6 星期六</td></tr></table></div>	星期几	0 星期日	1 星期一	2 星期二	3 星期三	4 星期四	5 星期五	6 星期六	S/U (请求)																	
星期几																													
0 星期日																													
1 星期一																													
2 星期二																													
3 星期三																													
4 星期四																													
5 星期五																													
6 星期六																													
SD220	LED 显示数据	显示指示器数据	<ul style="list-style-type: none">在错误发生时存储消息(16 字符的 ASCII 码)(包含报警器为 ON)。 <div><div>b15 到 b8</div><div>b7 到 b0</div><table><tr><td>SD220</td><td>从右边开始第15个字符</td><td>从右边开始第16个字符</td></tr><tr><td>SD221</td><td>从右边开始第13个字符</td><td>从右边开始第14个字符</td></tr><tr><td>SD222</td><td>从右边开始第11个字符</td><td>从右边开始第12个字符</td></tr><tr><td>SD223</td><td>从右边开始第9个字符</td><td>从右边开始第10个字符</td></tr><tr><td>SD224</td><td>从右边开始第7个字符</td><td>从右边开始第8个字符</td></tr><tr><td>SD225</td><td>从右边开始第5个字符</td><td>从右边开始第6个字符</td></tr><tr><td>SD226</td><td>从右边开始第3个字符</td><td>从右边开始第4个字符</td></tr><tr><td>SD227</td><td>从右边开始第1个字符</td><td>从右边开始第2个字符</td></tr></table></div>	SD220	从右边开始第15个字符	从右边开始第16个字符	SD221	从右边开始第13个字符	从右边开始第14个字符	SD222	从右边开始第11个字符	从右边开始第12个字符	SD223	从右边开始第9个字符	从右边开始第10个字符	SD224	从右边开始第7个字符	从右边开始第8个字符	SD225	从右边开始第5个字符	从右边开始第6个字符	SD226	从右边开始第3个字符	从右边开始第4个字符	SD227	从右边开始第1个字符	从右边开始第2个字符	S (当被改变时)	
SD220			从右边开始第15个字符	从右边开始第16个字符																									
SD221			从右边开始第13个字符	从右边开始第14个字符																									
SD222			从右边开始第11个字符	从右边开始第12个字符																									
SD223			从右边开始第9个字符	从右边开始第10个字符																									
SD224			从右边开始第7个字符	从右边开始第8个字符																									
SD225			从右边开始第5个字符	从右边开始第6个字符																									
SD226			从右边开始第3个字符	从右边开始第4个字符																									
SD227	从右边开始第1个字符	从右边开始第2个字符																											
SD221																													
SD222																													
SD223																													
SD224																													
SD225																													
SD226																													
SD227																													
SD240	基本模式	0:自动模式 1:详细模式	<ul style="list-style-type: none">存储此基本模式。	S (初始化)																									
SD241	扩展基板的号码	0:只有基本基板 1 到 4: 扩展基板号码	<ul style="list-style-type: none">存储被安装的扩展基板的最大号码。	S (初始化)																									
SD242	安装的 Q 基板存在/不存在	基板类型区别 0:基板没有被安装 1:Q * * B 被安装	<div><div>b4</div><div>b2 b1 b0</div><div>固定为 0</div><div>到</div><div>主基板</div><div>第1扩展基板</div><div>第2扩展基板</div><div>到</div><div>第4扩展基板</div></div>	S (初始化)																									
SD243	基板插槽号码 (操作状态)	基板插槽号码	<div><div>b15 到 b12</div><div>b11 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div><table><tr><td>SD243</td><td>扩展3</td><td>扩展2</td><td>扩展1</td><td>主</td></tr><tr><td>SD244</td><td colspan="3">固定为0</td><td>扩展4</td></tr></table></div>	SD243	扩展3	扩展2	扩展1	主	SD244	固定为0			扩展4	S (初始化)															
SD243			扩展3	扩展2	扩展1	主																							
SD244	固定为0			扩展4																									
SD244	<ul style="list-style-type: none">如上所示, 每个区域存储安装的插槽号码。 (当参数设置被执行时, 设定的插槽号码)																												

特殊寄存器列表(续表)

号码	名称	含义	解释	设定者 (当被设定时)	相应的 CPU
SD245	基板插槽号码 (安装状态)	基板插槽号码	<div><div><div>b15 到 b12</div><div>b11 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div><div>SD245</div><div>SD246</div></div><div><div>扩展3</div><div>扩展2</div><div>扩展1</div><div>主</div></div><div><div>固定为0</div><div>扩展4</div></div></div>	S(初始化)	系列号 04122 或更大
SD246			<ul style="list-style-type: none">如上所示, 每个区域存储了基板单元中每个模块安装的插槽号码(安装的基板单元插槽的实际号码)。		
SD250	安装的最大 I/O	安装的最大 I/O 号码	<ul style="list-style-type: none">当 SM250 从 OFF 变为 ON, 安装模块最后的 I/O 号码加 1 的高 2 位以 BIN 值的形式存储。	S(初始化)	○
SD254	MELSECNET/H 信息	安装模块的数量	<ul style="list-style-type: none">表示在 MELSECNET/H 上安装的模块的数量。	S(初始化)	
SD255		来自第一块模块的 I/O 号码	<ul style="list-style-type: none">安装的第一块模块的 MELSECNET/H I/O 号码		
SD256		来自第一块模块的网络号码	<ul style="list-style-type: none">安装的第一块模块的 MELSECNET/H 网络号码		
SD257		来自第一块模块的组号码	<ul style="list-style-type: none">安装的第一块模块的 MELSECNET/H 组号码		
SD258		来自第一块模块的站点号码	<ul style="list-style-type: none">安装的第一块模块的 MELSECNET/H 站点号码		
SD290	设备分配 (和参数内容相同)	分配给 X 的数据点数	<ul style="list-style-type: none">存储当前为 X 设备配置的数据点数	S(初始化)	
SD291		分配给 Y 的数据点数	<ul style="list-style-type: none">存储当前为 Y 设备配置的数据点数		
SD292		分配给 M 的数据点数	<ul style="list-style-type: none">存储当前为 M 设备配置的数据点数		
SD293		分配给 L 的数据点数	<ul style="list-style-type: none">存储当前为 L 设备配置的数据点数		
SD294		分配给 B 的数据点数	<ul style="list-style-type: none">存储当前为 B 设备配置的数据点数		
SD295		分配给 F 的数据点数	<ul style="list-style-type: none">存储当前为 F 设备配置的数据点数		
SD296		分配给 SB 的数据点数	<ul style="list-style-type: none">存储当前为 SB 设备配置的数据点数		
SD297		分配给 V 的数据点数	<ul style="list-style-type: none">存储当前为 V 设备配置的数据点数		
SD298	设备分配 (和参数内容相同)	分配给 S 的数据点数	<ul style="list-style-type: none">存储当前为 S 设备配置的数据点数	S(初始化)	
SD299		分配给 T 的数据点数	<ul style="list-style-type: none">存储当前为 T 设备配置的数据点数		
SD300		分配给 ST 的数据点数	<ul style="list-style-type: none">存储当前为 ST 设备配置的数据点数		
SD301	设备分配 (和参数内容相同)	分配给 C 的数据点数	<ul style="list-style-type: none">存储当前为 C 设备配置的数据点数	S(初始化)	
SD302	设备分配 (和参数内容相同)	分配给 D 的数据点数	<ul style="list-style-type: none">存储当前为 D 设备配置的数据点数		
SD303		分配给 W 的数据点数	<ul style="list-style-type: none">存储当前为 W 设备配置的数据点数		
SD304		分配给 SW 的数据点数	<ul style="list-style-type: none">存储当前为 SW 设备配置的数据点数		
SD315	为通讯处理保留的时间	为通讯处理保留的时间	<ul style="list-style-type: none">为 GX Developer 或其它单元的通讯处理保留的指定时间。指定的值越大, 和其它设备(GX Developer, 串行通讯单元)通讯的响应时间就变得越短。设定范围: 1 到 100ms如果指定的值超过了上述范围, 则被认为没有设置。扫描时间增加指定的时间值。	END 处理	
SD340	以太网信息	安装的模块号码	<ul style="list-style-type: none">表示安装在以太网上的模块数	S(初始化)	
SD341		I/O 号码	<ul style="list-style-type: none">表示安装的以太网模块的 I/O 号码。		
SD342		网络号码	<ul style="list-style-type: none">表示安装的以太网模块的网络号码。		
SD343		组号码	<ul style="list-style-type: none">表示安装的以太网模块的组号码。		
SD344		站点号码	<ul style="list-style-type: none">表示安装的以太网模块的站点号码。		

特殊寄存器列表(续表)

号码	名称	含义	解释	设定者 (当被设定时)	相应的 CPU	
SD393	多 CPU 系统信息	多 CPU 系统的 CPU 数量	● 组成多 CPU 系统的 CPU 模块数量被存储。(1 到 3, 空的也包括)	S(初始化)	○ 系列号 04122 或更大	
SD394		CPU 安装信息	● 不论 CPU 模块是否被安装, 第一到第三 CPU 的模块类型都被存储。 <div><div>b15 到 b12 b11 到 b8 b7 到 b4 b3 到 b0</div><div>SD394 空(0) PLC No. 3 PLC No. 2 PLC No. 1</div><div></div></div>			
SD395		多 CPU 号码	● 在多 PLC 组态中, 主机 PLC 的 PLC 号码被存储。 第 1 CPU: 1, 第 2 CPU: 2, 第 3 CPU: 3			
SD396		第 1 CPU 操作状态	● 每个 CPU 号码的操作信息被存储。 (关于在 SD393 中表示的多 PLC 号码的信息被存储)	S(当 END 处理错误发生时)		
SD397		第 2 CPU 操作状态	<div><div>b15 b14 到 b8 b7 to b4 b3 到 b0</div><div></div><div>安装或未安装 0:正常 0:RUN 0:未安装 1:监视故障 1:STEP RUN 1:安装 2:媒介故障 2:STOP </div></div>			

(3) 系统时钟/计数器

号码	名称	含义	解释	设定者 (当被设定时)	相应的 CPU
SD412	1 秒计数器	在 1 秒单元内计数的数目	<ul style="list-style-type: none"> 跟随可编程控制器 CPU 模块 RUN, 每秒加 1 计数在从 0 到 32767 到 32768 到 0 之间重复 	S (状态改变)	○
SD414	2n 秒时钟设置	2n 秒时钟单元	<ul style="list-style-type: none"> 存储 2n 秒时钟的值 n(默认为 30) 在 1 和 32767 之间设置。 	U	
SD420	扫描计数器	每个扫描内的计数数目	<ul style="list-style-type: none"> 在 CPU 模块被设置为 RUN 后, 每个扫描执行增加 1。* 计数在从 0 到 32767 到 32768 到 0 之间重复 	S(每个 END 处理)	

(4) 扫描信息

号码	名称	含义	解释	设定者 (当被设定时)	相应的 CPU
SD520	当前扫描时间	当前扫描时间 (在 1ms 内)	● 存储当前扫描时间(在 1ms 单位内) 范围从 0 到 65535	S(每个 END 处理)	○
SD521		当前扫描时间 (在 100μs 单元内)	● 存储当前扫描时间 (在 100μs 单位内) 范围从 00000 到 900 (例子) 当前扫描时间 23.6ms 被按照下列格式存储: D520=23 D521=600	S(每个 END 处理)	
SD524		最小扫描时间	最小扫描时间 (在 1ms 单位内)	● 存储扫描时间的最小值(在 1ms 单位内) ● 范围从 0 到 65535	
SD525	最小扫描时间 (在 100μs 单位内)		● 存储扫描时间的最小值(在 100μs 单位内) ● 范围从 000 到 900	S(每个 END 处理)	
SD526	最大扫描时间	最大扫描时间 (在 1ms 单位内)	● 存储扫描时间的最大值, 不包括第一个扫描周期(在 1ms 单位内) ● 范围从 0 到 65535	S(每个 END 处理)	
SD527		最大扫描时间 (在 100μs 单位内)	● 存储扫描时间的最大值, 不包括第一个扫描周期 (在 100μs 单位内) ● 范围从 000 到 900		

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定时)	相应的 CPU
SD540	END 处理时间	END 处理时间 (在 1ms 单元内)	● 存储从扫描程序完成后到下一个扫描启动之间的时间。(在 1ms 单元内) ● 范围从 0 到 65535	S(每个 END 处理)	○
SD541		END 处理时间 (在 100μs 单元内)	● 存储从扫描程序完成后到下一个扫描启动之间的时间。(在 100μs 单元内) ● 范围从 000 到 900		
SD542	固定扫描等待时间	固定扫描等待时间 (在 1ms 单元内)	● 当固定扫描时间被设定时存储等待时间。(在 1ms 单元内) ● 范围从 0 到 65535	S(第一个 END 处理)	
SD543		固定扫描等待时间 (在 100μs 单元内)	● 当固定扫描时间被设定时存储等待时间。(在 100μs 单元内) ● 范围从 000 到 900		
SD548	扫描程序执行时间	扫描程序执行时间 (在 1ms 单元内)	● 存储 1 个扫描周期内用于扫描执行类型程序的执行时间(在 1ms 单元内) ● 范围从 0 到 65535 ● 存储每个扫描	S(每个 END 处理)	
SD549		扫描程序执行时间 (在 100μs 单元内)	● 存储 1 个扫描周期内用于扫描执行类型程序的执行时间(在 100μs 单元内) ● 范围从 000 到 900 ● 存储每个扫描		

(5) 驱动信息

号码	名称	含义	解释	设定者 (当被设定时)	相应的 CPU
SD620	驱动器 3/4 模块	驱动器 3/4 模块	<div><div>● 表示驱动器 3/4 模块</div><div><div><div>b15 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div><div>0</div><div></div><div></div></div></div><div><div>驱动器3 (标准RAM)</div><div>0:不存在 1:存在</div></div><div><div>驱动器4 (标准ROM)</div><div>固定在“3”</div></div></div> <div>驱动器 4 被固定为“3”，因为它是内嵌在 Flash ROM 中</div>	S(初始化)	
SD622	驱动器 3(标准 RAM) 容量	驱动器 3 容量	<div><div>● 驱动器 3 容量以 1k 字节为单位存储的。 (固定为“61”，因为它有 61k 字节内嵌的 RAM)</div></div>	S(初始化)	
SD623	驱动器 4(标准 ROM) 容量	驱动器 4 容量	<div><div>● 驱动器 4 容量以 1k 字节为单位存储。</div></div>	S(初始化)	
SD624	驱动器 3 使用条件	驱动器 3 使用条件	<div><div>● 驱动器 3 使用条件被以位模式存储。</div><div><div><div>b15 到 b4</div><div>b3 到 b0</div></div><div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div></div><div>文件寄存器(R) 1:使用中 0:未被使用</div></div>	S (状态改变)	○
SD640	文件寄存器驱动	驱动器号码:	<div><div>● 存储正被文件寄存器使用的驱动器号码</div></div>	S(状态改变) * 1	
SD641	文件寄存器文件名称	文件寄存器文件名称	<div><div>● 以 ASCII 码存储在参数上选择的文件寄存器文件名称(带扩展)。</div><div><div><div>b15 到 b8</div><div>b7 到 b0</div></div><div><div>SD641 第二个字符(A)</div><div>第一个字符(M)</div></div><div><div>SD642 第四个字符(N)</div><div>第三个字符(I)</div></div><div><div>SD643 第六个字符()</div><div>第五个字符()</div></div><div><div>SD644 第八个字符()</div><div>第七个字符()</div></div><div><div>SD645 扩展的第一个字符(Q)</div><div>2Eh (.)</div></div><div><div>SD646 扩展的第三个字符(R)</div><div>扩展的第二个字符(D)</div></div></div></div>	S(初始化)	
SD642					
SD643					
SD644					
SD645					
SD646					
SD647	文件寄存器容量	文件寄存器容量	<div><div>● 以 1k 字为单元，存储当前选择的文件寄存器的数据容量容量。</div></div>	S(初始化)	
SD648	文件寄存器块号码	文件寄存器块号码	<div><div>● 存储当前选择的文件寄存器块号码。</div></div>	S(状态改变) * 1	

* 1: 在参数执行后，当 CPU 被停止，然后启动时，或者当 RSET 指令被执行时，此数据被置位。

(6) 和寄存器相关的指令

号码	名称	含义	解释	设定者 (当被设定时)	相应的 CPU																				
SD715	IMASK 指令掩码模式	掩码模式	<div>● IMASK 指令掩码模式按照下列样式存储:</div> <div><table><tr><td>b15</td><td>到</td><td>b1</td><td>b0</td></tr><tr><td>SD715</td><td>115</td><td>到</td><td>11 10</td></tr><tr><td>SD716</td><td>131</td><td>到</td><td>117 116</td></tr><tr><td>SD717</td><td>147</td><td>到</td><td>133 132</td></tr></table></div>	b15	到	b1	b0	SD715	115	到	11 10	SD716	131	到	117 116	SD717	147	到	133 132	S (执行过程中)	○				
b15				到	b1	b0																			
SD715				115	到	11 10																			
SD716	131	到	117 116																						
SD717	147	到	133 132																						
SD716																									
SD717																									
SD718	累加器	累加器	<div>● 用于 A 系列程序中累加器的代替者。</div>	S/U																					
SD719																									
SD774	PID 界限设置 (用于精确的微分)	0: 有界限 1: 无界限	<div>如下所示, 指定每个 PID 回路的界限。</div> <div><table><tr><td>b15</td><td>到</td><td>b8</td><td>b7</td><td>到</td><td>b1</td><td>b0</td></tr><tr><td>SD774</td><td colspan="2" style="background: linear-gradient(to right, black 49%, white 49% 51%, black 51% 53%);"></td><td>回路8</td><td>到</td><td>回路2</td><td>回路1</td></tr></table></div>	b15	到	b8	b7	到	b1	b0	SD774			回路8	到	回路2	回路1	U							
b15	到	b8	b7	到	b1	b0																			
SD774			回路8	到	回路2	回路1																			
SD778	在 COM 指令执行时的刷新处理选择	<div>b0 到 b4 (默认: 0) 0: 刷新未被执行 1: 刷新被执行 b15 0: 执行一般数据处理 1: 不执行一般数据处理</div>	<div><div>● 在 COM 指令被执行时, 选择每个刷新处理是否被执行。</div><div>● 当 SM775 变为 ON 时, SD778 的指定变为有效。</div></div> <div><table><tr><td>b15</td><td>b14</td><td>到</td><td>b5</td><td>b4</td><td>b3</td><td>b2</td><td>b1</td><td>b0</td></tr><tr><td>SD778</td><td>1/0</td><td>0</td><td>1/0</td><td>1/0</td><td>1/0</td><td>1/0</td><td>1/0</td><td>1/0</td></tr></table><div><div>I/O刷新</div><div>CC-Link刷新</div><div>MELSECNET/H刷新</div><div>智能功能模块的自动刷新</div><div>CPU共享存储器的自动刷新</div><div>执行/不执行一般数据处理</div></div></div>	b15	b14	到	b5	b4	b3	b2	b1	b0	SD778	1/0	0	1/0	1/0	1/0	1/0	1/0	1/0	U	○ 系列号 04122 或更大		
b15	b14	到	b5	b4	b3	b2	b1	b0																	
SD778	1/0	0	1/0	1/0	1/0	1/0	1/0	1/0																	
SD781 T0 SD785	IMASK 指令的掩码模式	掩码模式	<div>● IMASK 指令掩码模式按照下列样式存储:</div> <div><table><tr><td>b15</td><td>到</td><td>b1</td><td>b0</td></tr><tr><td>SD781</td><td>163</td><td>到</td><td>149 148</td></tr><tr><td>SD782</td><td>179</td><td>到</td><td>165 164</td></tr><tr><td colspan="4">到</td></tr><tr><td>SD785</td><td>1127</td><td>到</td><td>1113 1112</td></tr></table></div>	b15	到	b1	b0	SD781	163	到	149 148	SD782	179	到	165 164	到				SD785	1127	到	1113 1112	S (执行过程中)	○
b15	到	b1	b0																						
SD781	163	到	149 148																						
SD782	179	到	165 164																						
到																									
SD785	1127	到	1113 1112																						
SD794	PID 界限设置 (用于精确的微分)	0: 有界限 1: 无界限	<div>如下所示, 指定每个 PID 回路的界限。</div> <div><table><tr><td>b15</td><td>到</td><td>b8</td><td>b7</td><td>到</td><td>b1</td><td>b0</td></tr><tr><td>SD794</td><td colspan="2" style="background: linear-gradient(to right, black 49%, white 49% 51%, black 51% 53%);"></td><td>回路8</td><td>到</td><td>回路2</td><td>回路1</td></tr></table></div>	b15	到	b8	b7	到	b1	b0	SD794			回路8	到	回路2	回路1	U	○ 系列号 04122 或更大						
b15	到	b8	b7	到	b1	b0																			
SD794			回路8	到	回路2	回路1																			

附录 4.2 高性能型 QCPU/QnACPU 的特殊寄存器列表

特殊寄存器，SD，是应用固化在 PLC 中的内部寄存器。
 由于这个原因，它们不能以普通的寄存器那样的方式被顺控程序使用。
 然而，数据可以按照需要被写入以控制 CPU 模块和远程 I/O 模块。
 如果没有相反的特殊指定，数据以 BIN 值的形式存储在特殊寄存器内。

下表中标题的含义如下所示。

项目	项目的功能
号码	● 表示特殊继电器的号码
名称	● 表示特殊继电器的名称
含义	● 表示特殊寄存器的内容
解释	● 包含特殊继电器内容的详细信息。
设置者(当被设置时)	● 表示继电器是被系统设置还是被用户设置，并且，如果是被系统设置，当设置被执行时。 <设置者> S : 由系统设置 U : 由用户设置(GX Developer 的顺序程序或者测试操作) S/U : 由系统和用户设置 <当被设置时> → 只有当寄存器由系统设置时才被指示 每个 END : 在每条 END 处理过程中被设置 初始化 : 只在初始化处理过程中被设置(当供电电源变为 ON, 或当系统由 STOP 变为 RUN 时) 状态改变 : 只有在状态有改变时才被设置 错误 : 在有错误产生时才被设置 指令执行 : 在指令被执行时被设置 请求 : 只有在有用户请求时才被设置(通过 SM, 等)
相应的 ACPU M9 □ □ □	● 表示 ACPU 中相应的特殊寄存器(D9 □ □ □)(内容有改变时的改变和符号) ● “新增”表示项目是新近增加到高性能模块 QCPU/QnACPU 中的。
相应的 CPU	● 表示相应的 CPU 模块类型名称 ○+Rem: 可以被应用到所有的 CPU 类型和 MELSECNET/H 远程 I/O 模块。 ○: 可以被应用到所有类型的 CPU 模块 QCPU: 可以被应用到高性能模块 QCPU。 QnA: 可以被应用到 QnA 系列和 Q2ACPU 系列 Remote: 可以被应用到 MELSECNET/H 远程 I/O 模块 每个 CPU 类型名称: 只能被应用到指定的 CPU。(例如 Q4ARCPU, Q3ACPU)

对于下列项目的详细信息，参见下列手册：

- 网络 → ● 用于和 Q 相应的 MELSECNET/H 网络系统参考手册(PLC 至 PLC 网络)
 - 用于和 Q 相应的 MELSECNET/H 网络系统参考手册(远程 I/O 网络)
 - 用于 QnA/Q4AR MELSECNET/10 网络系统参考手册
- SFC → QCPU(Q 模式)/QnACPU 编程手册(SFC)

要点
(1) SD1200 和 SD1255 被用于 QnACPU。 高性能模块 QCPU 中这些继电器是空白的。
(2) 特殊寄存器 SD1500 和其以后的专用于 Q4ARCPU。

特殊寄存器列表

(1) 诊断信息

号码	名称	含义	解释	设定者 (当被设定)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD0	诊断错误	诊断错误代码	<ul style="list-style-type: none"> 由诊断发现的错误代码存储成 BIN 数据。 内容和最新故障历史信息相同。 	S(错误)	D9008 格式改变	
SD1	诊断错误发生时的时钟时间	诊断错误发生时的时钟时间	<ul style="list-style-type: none"> SD0 数据更新时的年(后两位数)和月被存储成 BCD 2 位代码。 (例子) <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">b15 到 b8 b7 到 b0 年(0到99) 月(1到12)</div> <div style="text-align: left;">:十月, 1995 9510H</div> </div> 	S(错误)	新增	
SD2			<ul style="list-style-type: none"> SD0 被更新时的日期和小时被存储成 BCD 2 位代码。 (例子) <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">b15 到 b8 b7 到 b0 日期(1到31) 小时(0到23)</div> <div style="text-align: left;">:10下午, 25号 2510H</div> </div> 			
SD3			<ul style="list-style-type: none"> SD0 被更新时的分钟和秒被存储成 BCD 2 位代码。 (例子) <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">b15 到 b8 b7 到 b0 分钟(0到59) 秒(0到59)</div> <div style="text-align: left;">:35分钟48秒 (小时后过了) 3548H</div> </div> 			
SD4	错误信息分类	错误信息分类代码	<ul style="list-style-type: none"> 在公用信息区域(SD5 到 SD15)和单独信息区域(SD16 到 SD26)内, 用于帮助标志哪种类型的信息被存储的分类代码存储在这里。 <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">b15 到 b8 b7 到 b0 单独信息分类代码 公用信息分类代码</div> </div> 公用信息分类代码存储了下列代码: <ul style="list-style-type: none"> 0: 无错误 1: 单元/模块号码/PLC 号码/基板号码 * 2: 文件名称/驱动器名称 3: 时间(设定值) 4: 程序错误定位 5: 切换原因(只用于 Q4AR) <p style="margin-left: 20px;">*: 对于一个多 PLC 系统, 模块号码或 PLC 号码依据发生的错误存储。 (指向存储的号码对应的错误代码) 1 号 PLC: 1, 2 号 PLC: 2, 3 号 PLC: 3, 4 号 PLC: 4</p> 单独信息分类代码存储了下列代码: <ul style="list-style-type: none"> 0: 无错误 1: (打开) 2: 文件名称/驱动器名称 3: 时间(实际测量值) 4: 程序错误定位 5: 参数号码 6: 报警器号码 7: 检查指令故障号码 	S(错误)	新增	○+Rem

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU		
SD5	错误公用信息	错误公用信息	<ul style="list-style-type: none">● 和错误代码(SD0)相应的普通信息存储在这里。● 接下来的四字节的信息存储在这里：<ul style="list-style-type: none">① 插槽号码	S(错误)	新增	○+Rem		
SD6								
SD7								
SD8								
SD9								
SD10								
SD11								
SD12								
SD13								
SD14								
SD15								
SD5			含义				插槽号/PLC号码/基板号码*1*2	(未被用作基板号码)
SD6			I/O号码					
SD7			(空白)					
SD8								
SD9								
SD10								
SD11								
SD12								
SD13								
SD14								
SD15								
*1: 对于多 CPU 系统, 插槽号码或 CPU 号码依据发生的错误存储。 多 CPU 系统中的 Slot 0 是指, 最右边 CPU 模块中右边的插槽。 (指向被存储号码对应的错误代码。) 第 1 CPU: 1, 第 2 CPU: 2, 第 3 CPU: 3, 第 4 CPU: 4								
*2: 如果在 MELSECNET/H 远程 I/O 站点安装的模块中, 发生保险丝 烧坏或者 I/O 校验错误, 网络号码被存储到高 8 位, 站点号码 被存储到低 8 位。 使用 I/O 号码去检查保险丝烧坏或者发生 I/O 校验错误的模 块。								
② 文件名称/驱动器名称								
号码	含义	文件名称=						
SD5	驱动器	ABCDEFGH, IJK						
SD6	文件名称 (ASCII码: 8字符)	b15 到 b8 b7 到 b0						
SD7		42H(B) 41H(A)						
SD8		44H(D) 43H(C)						
SD9		46H(F) 45H(E)						
SD10	扩展*3	2EH(.)						
SD11	(ASCII码: 3字符)	48H(H) 47H(G)						
SD12	(空白)	49H(I) 2EH(.)						
SD13		4BH(K) 4AH(J)						
SD14								
SD15								

*3: 参考备注

备注

1) 扩展型如下所示。

SD10 高 8 位	SD11		扩展名称	文件类型
	低 8 位	高 8 位		
51H	50H	41H	QPA	参数
51H	50H	47H	QPG	顺序程序/SFC 程序
51H	43H	44H	QCD	设备注释
51H	44H	49H	QDI	设备初始值
51H	44H	52H	QDR	文件寄存器
51H	44H	53H	QDS	仿真数据
51H	44H	4CH	QDL	本地设备
51H	54H	53H	QTS	采样跟踪数据(用于 QnA)
51H	54H	4CH	QTL	状态闭锁数据(用于 QnA)
51H	54H	50H	QTP	程序跟踪数据(用于 QnA)
51H	54H	52H	QTR	SFC 跟踪文件
51H	46H	44H	QFD	故障历史数据

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定 定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU																									
SD5	错误公用信息	错误公用信息	③ 时间(设定值)	S(错误)	新增	○+Rem																									
SD6																															
SD7																															
SD8																															
SD9																															
SD10																															
SD11																															
SD12																															
SD13																															
SD14																															
SD15																															
<table><tr><td>号码</td><td>含义</td></tr><tr><td>SD5</td><td>时间:1μs单位(0到999μs)</td></tr><tr><td>SD6</td><td>时间:1ms单位(0到65535ms)</td></tr><tr><td>SD7</td><td rowspan="10">(空白)</td></tr><tr><td>SD8</td></tr><tr><td>SD9</td></tr><tr><td>SD10</td></tr><tr><td>SD11</td></tr><tr><td>SD12</td></tr><tr><td>SD13</td></tr><tr><td>SD14</td></tr><tr><td>SD15</td></tr></table>							号码	含义	SD5	时间:1μs单位(0到999μs)	SD6	时间:1ms单位(0到65535ms)	SD7	(空白)	SD8	SD9	SD10	SD11	SD12	SD13	SD14	SD15									
号码			含义																												
SD5			时间:1μs单位(0到999μs)																												
SD6			时间:1ms单位(0到65535ms)																												
SD7	(空白)																														
SD8																															
SD9																															
SD10																															
SD11																															
SD12																															
SD13																															
SD14																															
SD15																															
④ 程序错误定位																															
<table><tr><td>号码</td><td>含义</td></tr><tr><td>SD5</td><td rowspan="4">文件名称 (ASCII码:8字符)</td></tr><tr><td>SD6</td></tr><tr><td>SD7</td></tr><tr><td>SD8</td></tr><tr><td>SD9</td><td>扩展</td><td>2EH(.)</td></tr><tr><td>SD10</td><td colspan="2">(ASCII码:3字符)</td></tr><tr><td>SD11</td><td colspan="2">样式*4</td></tr><tr><td>SD12</td><td colspan="2">块号码</td></tr><tr><td>SD13</td><td colspan="2">步号码/转换号码</td></tr><tr><td>SD14</td><td colspan="2">顺序步号码(L)</td></tr><tr><td>SD15</td><td colspan="2">顺序步号码(H)</td></tr></table>				号码	含义	SD5	文件名称 (ASCII码:8字符)	SD6	SD7	SD8	SD9	扩展	2EH(.)	SD10	(ASCII码:3字符)		SD11	样式*4		SD12	块号码		SD13	步号码/转换号码		SD14	顺序步号码(L)		SD15	顺序步号码(H)	
号码	含义																														
SD5	文件名称 (ASCII码:8字符)																														
SD6																															
SD7																															
SD8																															
SD9	扩展	2EH(.)																													
SD10	(ASCII码:3字符)																														
SD11	样式*4																														
SD12	块号码																														
SD13	步号码/转换号码																														
SD14	顺序步号码(L)																														
SD15	顺序步号码(H)																														
* 4 模式数据的内容																															
<table><tr><td>15</td><td>14</td><td>到</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>←(位号码)</td></tr><tr><td>0</td><td>0</td><td>to</td><td>0</td><td>0</td><td>*</td><td>*</td><td>*</td><td></td></tr></table> <p>(未被使用)</p> <p>└── SFC块标志存在(1)/不存在(0)</p> <p>└── SFC步标志存在(1)/不存在(0)</p> <p>└── SFC 转换标志存在(1)/不存在(0)</p>				15	14	到	4	3	2	1	0	←(位号码)	0	0	to	0	0	*	*	*											
15	14	到	4	3	2	1	0	←(位号码)																							
0	0	to	0	0	*	*	*																								
SD15			⑤ 切换原因	S(错误)	新增	Q4AR																									
<table><tr><td>号码</td><td>含义</td></tr><tr><td>SD5</td><td>切换原因(0:自动切换/1:手动切换)</td></tr><tr><td>SD6</td><td>切换方向(0:备用系统切换到控制系统/ 1: 控制系统切换到备用系统)</td></tr><tr><td>SD7</td><td rowspan="5">跟踪标志*5</td></tr><tr><td>SD8</td></tr><tr><td>SD9</td></tr><tr><td>SD10</td></tr><tr><td>SD11</td></tr><tr><td>SD12</td><td rowspan="4">(空白)</td></tr><tr><td>SD13</td></tr><tr><td>SD14</td></tr><tr><td>SD15</td></tr></table>				号码	含义	SD5	切换原因(0:自动切换/1:手动切换)	SD6	切换方向(0:备用系统切换到控制系统/ 1: 控制系统切换到备用系统)	SD7	跟踪标志*5	SD8	SD9	SD10	SD11	SD12	(空白)	SD13	SD14	SD15											
号码	含义																														
SD5	切换原因(0:自动切换/1:手动切换)																														
SD6	切换方向(0:备用系统切换到控制系统/ 1: 控制系统切换到备用系统)																														
SD7	跟踪标志*5																														
SD8																															
SD9																															
SD10																															
SD11																															
SD12	(空白)																														
SD13																															
SD14																															
SD15																															
* 5 跟踪标志内容																															
表明跟踪数据是否有效。																															
<table><tr><td>15</td><td>14</td><td>到</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>←(位号码)</td></tr><tr><td>0</td><td>0</td><td>到</td><td>0</td><td>0</td><td>*</td><td>*</td><td>*</td><td></td></tr></table> <p>(未被使用)</p> <p>└── 无效工作数据 无效0/有效(1)</p> <p>└── 系统数据 (SFC活动步信息) 无效(0)/有效(1)</p> <p>└── 切换原因无效(0)/ 有效(1)</p>				15	14	到	4	3	2	1	0	←(位号码)	0	0	到	0	0	*	*	*											
15	14	到	4	3	2	1	0	←(位号码)																							
0	0	到	0	0	*	*	*																								

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定 定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU																																																																																																																					
SD16	错误单独信息	错误单独信息	<div>● 和错误代码(SD0)相应的单独信息存储在这里。</div> <div>① 文件名/驱动器名</div> <div>(例子)</div> <table><tr><th>号码</th><th>含义</th><th>文件名称=</th></tr><tr><td>SD16</td><td>驱动器</td><td>ABCDEFGH, IJK</td></tr><tr><td>SD17</td><td rowspan="4">文件名称 (ASCII码:8字符)</td><td>b15 到 b8 b7 到 b0</td></tr><tr><td>SD18</td><td>42H(B) 41H(A)</td></tr><tr><td>SD19</td><td>44H(D) 43H(C)</td></tr><tr><td>SD20</td><td>46H(F) 45H(E)</td></tr><tr><td>SD21</td><td>扩展</td><td>2EH(.)</td></tr><tr><td>SD22</td><td rowspan="4">(ASCII码:3字符)</td><td>48H(H) 47H(G)</td></tr><tr><td>SD23</td><td>49H(I) 2EH(.)</td></tr><tr><td>SD24</td><td>4BH(K) 4AH(J)</td></tr><tr><td>SD25</td><td></td></tr><tr><td>SD26</td><td>(空白)</td><td></td></tr></table> <div>② 时间(实际测量值)</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td>时间:1μs单位(0到999μs)</td></tr><tr><td>SD17</td><td>时间:1ms单位(0到65535ms)</td></tr><tr><td>SD18</td><td rowspan="8">(空白)</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td></tr><tr><td>SD21</td></tr><tr><td>SD22</td></tr><tr><td>SD23</td></tr><tr><td>SD24</td></tr><tr><td>SD25</td></tr><tr><td>SD26</td></tr></table> <div>③ 程序错误定位</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td rowspan="4">文件名 (ASCII码:8字符)</td></tr><tr><td>SD17</td></tr><tr><td>SD18</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td><td>扩展</td></tr><tr><td>SD21</td><td>2EH(.)</td></tr><tr><td>SD22</td><td>(ASCII码:3字符)</td></tr><tr><td>SD23</td><td>样式*</td></tr><tr><td>SD24</td><td>块号码</td></tr><tr><td>SD25</td><td>步号码/转换号码</td></tr><tr><td>SD26</td><td>顺序步号码(L) 顺序步号码(H)</td></tr></table> <div>* 模式数据的内容</div> <table><tr><td>15 14 到 4 3 2 1 0</td><td>← (位号码)</td></tr><tr><td>0 0 到 0 0 * * *</td><td></td></tr></table> <div>(未使用)</div> <div>SFC块标志存在(1)/不存在(0)</div> <div>SFC步标志存在(1)/不存在(0)</div> <div>SFC 转换标志存在(1)/不存在(0)</div> <div>④ 参数号码</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td>参数号码*6</td></tr><tr><td>SD17</td><td rowspan="10">(空白)</td></tr><tr><td>SD18</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td></tr><tr><td>SD21</td></tr><tr><td>SD22</td></tr><tr><td>SD23</td></tr><tr><td>SD24</td></tr><tr><td>SD25</td></tr><tr><td>SD26</td></tr></table> <div>⑤ 报警器号码/CHK 指 令故障号码</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td>号码</td></tr><tr><td>SD17</td><td rowspan="10">(空白)</td></tr><tr><td>SD18</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td></tr><tr><td>SD21</td></tr><tr><td>SD22</td></tr><tr><td>SD23</td></tr><tr><td>SD24</td></tr><tr><td>SD25</td></tr><tr><td>SD26</td></tr></table> <div>⑥ 智能功能模块参数 错误 (只用于 QCPU)</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td>参数号码*6</td></tr><tr><td>SD17</td><td>智能功能模块的错误 编码</td></tr><tr><td>SD18</td><td rowspan="10">(空白)</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td></tr><tr><td>SD21</td></tr><tr><td>SD22</td></tr><tr><td>SD23</td></tr><tr><td>SD24</td></tr><tr><td>SD25</td></tr><tr><td>SD26</td></tr></table> <div>* 参数号码的详细信息, 请参考所使用 CPU 模块的用户手册。</div>	号码	含义	文件名称=	SD16	驱动器	ABCDEFGH, IJK	SD17	文件名称 (ASCII码:8字符)	b15 到 b8 b7 到 b0	SD18	42H(B) 41H(A)	SD19	44H(D) 43H(C)	SD20	46H(F) 45H(E)	SD21	扩展	2EH(.)	SD22	(ASCII码:3字符)	48H(H) 47H(G)	SD23	49H(I) 2EH(.)	SD24	4BH(K) 4AH(J)	SD25		SD26	(空白)		号码	含义	SD16	时间:1μs单位(0到999μs)	SD17	时间:1ms单位(0到65535ms)	SD18	(空白)	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26	号码	含义	SD16	文件名 (ASCII码:8字符)	SD17	SD18	SD19	SD20	扩展	SD21	2EH(.)	SD22	(ASCII码:3字符)	SD23	样式*	SD24	块号码	SD25	步号码/转换号码	SD26	顺序步号码(L) 顺序步号码(H)	15 14 到 4 3 2 1 0	← (位号码)	0 0 到 0 0 * * *		号码	含义	SD16	参数号码*6	SD17	(空白)	SD18	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26	号码	含义	SD16	号码	SD17	(空白)	SD18	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26	号码	含义	SD16	参数号码*6	SD17	智能功能模块的错误 编码	SD18	(空白)	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26	S(错误)	新增	○ +Rem
号码			含义	文件名称=																																																																																																																							
SD16			驱动器	ABCDEFGH, IJK																																																																																																																							
SD17			文件名称 (ASCII码:8字符)	b15 到 b8 b7 到 b0																																																																																																																							
SD18				42H(B) 41H(A)																																																																																																																							
SD19				44H(D) 43H(C)																																																																																																																							
SD20				46H(F) 45H(E)																																																																																																																							
SD21			扩展	2EH(.)																																																																																																																							
SD22			(ASCII码:3字符)	48H(H) 47H(G)																																																																																																																							
SD23				49H(I) 2EH(.)																																																																																																																							
SD24				4BH(K) 4AH(J)																																																																																																																							
SD25																																																																																																																											
SD26			(空白)																																																																																																																								
号码			含义																																																																																																																								
SD16			时间:1μs单位(0到999μs)																																																																																																																								
SD17			时间:1ms单位(0到65535ms)																																																																																																																								
SD18			(空白)																																																																																																																								
SD19																																																																																																																											
SD20																																																																																																																											
SD21																																																																																																																											
SD22																																																																																																																											
SD23																																																																																																																											
SD24																																																																																																																											
SD25																																																																																																																											
SD26																																																																																																																											
号码			含义																																																																																																																								
SD16	文件名 (ASCII码:8字符)																																																																																																																										
SD17																																																																																																																											
SD18																																																																																																																											
SD19																																																																																																																											
SD20	扩展																																																																																																																										
SD21	2EH(.)																																																																																																																										
SD22	(ASCII码:3字符)																																																																																																																										
SD23	样式*																																																																																																																										
SD24	块号码																																																																																																																										
SD25	步号码/转换号码																																																																																																																										
SD26	顺序步号码(L) 顺序步号码(H)																																																																																																																										
15 14 到 4 3 2 1 0	← (位号码)																																																																																																																										
0 0 到 0 0 * * *																																																																																																																											
号码	含义																																																																																																																										
SD16	参数号码*6																																																																																																																										
SD17	(空白)																																																																																																																										
SD18																																																																																																																											
SD19																																																																																																																											
SD20																																																																																																																											
SD21																																																																																																																											
SD22																																																																																																																											
SD23																																																																																																																											
SD24																																																																																																																											
SD25																																																																																																																											
SD26																																																																																																																											
号码	含义																																																																																																																										
SD16	号码																																																																																																																										
SD17	(空白)																																																																																																																										
SD18																																																																																																																											
SD19																																																																																																																											
SD20																																																																																																																											
SD21																																																																																																																											
SD22																																																																																																																											
SD23																																																																																																																											
SD24																																																																																																																											
SD25																																																																																																																											
SD26																																																																																																																											
号码	含义																																																																																																																										
SD16	参数号码*6																																																																																																																										
SD17	智能功能模块的错误 编码																																																																																																																										
SD18	(空白)																																																																																																																										
SD19																																																																																																																											
SD20																																																																																																																											
SD21																																																																																																																											
SD22																																																																																																																											
SD23																																																																																																																											
SD24																																																																																																																											
SD25																																																																																																																											
SD26																																																																																																																											
SD26																																																																																																																											

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD50	错误复位	执行错误复位的错误号码	<ul style="list-style-type: none"> ● 存储执行错误复位的错误号码 	U	新增	○+Rem
SD51	电池低闭锁	位模式表示了哪里发生了电池电压下降	<ul style="list-style-type: none"> ● 当电池电压下降时，所有对应的位变为 1 (ON)。 ● 随后，即使电池电压恢复为正常，这些位也保持为 1 (ON)。  <ul style="list-style-type: none"> ● 当电池电压低发生时，这些报警数据可以在指定的时间内保持。 ● 错误表示电池放电结束。 ● 当 QCPU 被使用时，存储卡 B 是标准的，这样相应的位就保持常 OFF。 	S (错误)	新增	○
SD52	电池低	位模式表示了哪里发生了电池电压下降	<ul style="list-style-type: none"> ● 和上面的 SD51 的配置一样 ● 变成 0 (OFF)，当电池电压此后恢复为正常。 ● 当 QCPU 被使用时，存储卡 B 是标准的，这样相应的位就保持常 OFF。 	S (错误)	新增	
SD53	AC/DC DOWN 检测	AC/DC DOWN 的次数	<ul style="list-style-type: none"> ● 在 CPU 模块的计算中，每次输入电压下降到额定值的 85% (AC 电源) / 65% (DC 电源 r)，此值就加 1 并以 BIN 码形式存储。 	S (错误)	D9005	○+Rem
SD54	MINI 链接错误	错误探测状态	<p>① 当安装好的 MINI (S3) 中的任何一个 X(n+0)/X(n+20)，X(n+6)/X(n+26)，X(n+7)/X(n+27) 和 X(n+8)/X(n+28) 变为 ON，相应站点的位就会变成 1 (ON)。</p> <p>② 变为 1 (ON)，当安装的 MINI (S3) 和 CPU 模块之间的通信不能建立时。</p> 	S (错误)	D9004 格式改变	QnA
SD60	烧坏的保险丝号码	有烧坏的保险丝的模块号码	<ul style="list-style-type: none"> ● 存储在这里的值，是有烧坏的保险丝的模块中最低站点 I/O 的号码。 	S (错误)	D9000	○+Rem
SD61	I/O 模块校验错误号码	I/O 模块校验错误模块号码	<ul style="list-style-type: none"> ● 发生 I/O 模块校验错误的模块中，模块的最低 I/O 号码。 	S (错误)	D9002	
SD62	报警器号码	报警器号码	<ul style="list-style-type: none"> ● 探测到的第一个报警器号码 (F 号码) 存储在这里。 	S (指令执行)	D9009	○
SD63	报警器数目	报警器数目	<ul style="list-style-type: none"> ● 存储搜索到的报警器数目。 	S (指令执行)	D9124	

特殊寄存器列表(续表)

号码	名称	含义	解释	设定者 (当被设定 时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU	
SD64	探测到的报警 器号码表	报警器探测号 码	<p>当 F 基于 OUT F 或 SET F 变为 ON 时, 从 SD64 到 SD79 逐渐地变为 ON 的 F 号码被登记。</p> <p>被 RST F 变为 OFF 的 F 号码被从 SD64 SD79 中删除, 并且在被删除的 F 号码移位到前面的寄存器后其它 F 号码被存储。</p> <p>LEDR 指令的执行将 SD64 到 SD79 中的内容一个接一个的移位。 (这也可以通过使用 Q3A/Q4ACPU 上的 INDICATOR RESET 开关来实现。)</p> <p>16 个报警器被探测到之后, 第 17 个探测不会被存储进 SD64 到 SD79 中。</p> <div><div>SET SET SET RST SET SET SET SET SET SET SET F50 F25 F99 F25 F15 F70 F65 F38 F110F151F210 LEDR</div><div><div><div>SD62</div><div>0</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>99</div><div>...</div><div>(探测到的 号码)</div></div><div><div>SD63</div><div>0</div><div>1</div><div>2</div><div>3</div><div>2</div><div>3</div><div>4</div><div>5</div><div>6</div><div>7</div><div>8</div><div>9</div><div>8</div><div>...</div><div>(探测到的报 警器号码)</div></div></div><div><div><div>SD64</div><div>0</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>99</div></div><div><div>SD65</div><div>0</div><div>0</div><div>25</div><div>25</div><div>99</div><div>99</div><div>99</div><div>99</div><div>99</div><div>99</div><div>99</div><div>99</div><div>15</div></div><div><div>SD66</div><div>0</div><div>0</div><div>0</div><div>99</div><div>0</div><div>15</div><div>15</div><div>15</div><div>15</div><div>15</div><div>15</div><div>15</div><div>70</div></div><div><div>SD67</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>70</div><div>70</div><div>70</div><div>70</div><div>70</div><div>70</div><div>65</div></div><div><div>SD68</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>65</div><div>65</div><div>65</div><div>65</div><div>65</div><div>65</div><div>38</div></div><div><div>SD69</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>38</div><div>38</div><div>38</div><div>38</div><div>38</div><div>110</div></div><div><div>SD70</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>110</div><div>110</div><div>110</div><div>110</div><div>151</div></div><div><div>SD71</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>151</div><div>151</div><div>210</div></div><div><div>SD72</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>210</div><div>0</div></div><div><div>SD73</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div><div><div>SD74</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div><div><div>SD75</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div><div><div>SD76</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div><div><div>SD77</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div><div><div>SD78</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div><div><div>SD79</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div></div></div> <div>(探测到的 号码)</div>	S (指令执行)	D9125		
SD65					D9126		
SD66					D9127		
SD67					D9128		
SD68					D9129		
SD69					D9130		
SD70					D9131		
SD71					D9132		
SD72					新增		
SD73					新增		
SD74					新增		
SD75					新增		
SD76	新增						
SD77	新增						
SD78	新增						
SD79	新增						
SD80	CHK 号码	CHK 号码	● 由 CHK 指令探测到错误代码以 BCD 码存储。	S (指令执行)	新增		
SD90	步转换看门狗 定时器设定值 (只有在 SFC 程 序存在时才被 使能)	用于定时器设 置的 F 号码和 定时器溢出错 误	相当于 SM90	<div><div>● 设置当步转换看门狗定时器设置或看门狗定时器界限错误发生时变为 ON 的报警器号码(F 号码)。</div><div><div>b15 到 b8 b7 到 b0</div><div><div></div><div></div></div><div>F 号码设置 (0到255)</div><div>定时器时间界限 设置 (1到255 s: (1s单元))</div></div><div><div>● 在活动步启动定时器的过程中将 SM90 到 SM99 中的任一个变为 ON, 并且如果相应步接下来的转换条件不满足定时器的时间界限, 设置的报警器(F)变为 ON。</div></div></div>	U	D9108	
SD91			相当于 SM91			D9109	
SD92			相当于 SM92			D9110	
SD93			相当于 SM93			D9111	
SD94			相当于 SM94			D9112	
SD95			相当于 SM95			D9113	
SD96			相当于 SM96			D9114	
SD97			相当于 SM97			新增	
SD98			相当于 SM98			新增	
SD99			相当于 SM99			新增	
SD105	CH1 传送速度 设置(RS232)	当使用 GX Developer 时, 存储预设的传 送速度。	3 :300bps, 6 :600bps, 24 :2400bps, 48:4800bps 96 :9600bps, 192 :19.2kbps, 384 :38.4kbps 576 :57.6kbps, 1152 :115.2kbps	S	新增	QCPU 远程	

特殊寄存器列表

(2) 系统信息

号码	名称	含义	解释	设定者 (当被设定 定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD200	开关状态	CPU 开关状态	<div>● 远程 I/O 模块的开关状态按照下列格式存储。</div> <div><div><div>b15到b4</div><div>b3到b0</div></div><div>空白</div><div>①</div></div> <div>① 远程 I/O 模块开关状态常 1: STOP</div>	S(一直)	新增	远程
			<div>● CPU 开关状态按照下列格式存储:</div> <div><div><div>b15到b12</div><div>b11到b8</div><div>b7到b4</div><div>b3到b0</div></div><div>③空白②①</div></div> <div><div>①: CPU 开关状态0: RUN1: STOP2: L. CLR</div><div>②: 存储卡开关常 OFF</div><div>③: DIP 开关b8 到 b12 对应于系统设置开关 1 中的 SW1 到 SW5。0: OFF, 1: ONb13 到 b15 为空白。</div></div>	S(每个 END 处理)	新增	QCPU
			<div>● CPU 开关状态按照下列格式存储:</div> <div><div><div>b15到b12</div><div>b11到b8</div><div>b7到b4</div><div>b3到b0</div></div><div>③空白②①</div></div> <div><div>①: CPU 钥匙开关状态0: RUN1: STOP2: L. CLR</div><div>②: 存储卡开关b4 对应于存储卡 A, b5 对应于存储卡 B在 0 时为 OFF; 在 1 时为 ON</div><div>③: DIP 开关b8 到 b12 对应于系统设置开关 1 中的 SW1 到 SW5。b14 和 b15 分别对应系统设置开关 2 中的 SW1 和 SW2在 0 时为 OFF; 在 1 时为 ON</div></div>	S(每个 END 处理)	新增	QnA

特殊寄存器列表 (续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD201	LED 状态	CPU LED 的状态	<div>● 下列位模式被用来存储 CPU 模块上 LED 的状态：</div> <div><div><div>b15 到 b12</div><div>b11 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div><div>⑧</div><div>⑦</div><div>⑥</div><div>⑤</div><div>④</div><div>③</div><div>②</div><div>①</div></div></div> <div><div><div>①: RUN</div><div>②: ERR.</div><div>③: USER</div><div>④: BAT.</div></div><div><div>⑤: BOOT</div><div>⑥: 空白</div><div>⑦: 空白</div><div>⑧: MODE</div></div></div> <div>用于 MODE 的位模式 0: OFF, 1: 绿, 2: 黄</div>	S (状态改变)	新增	QCPU
			<div>● 关于 CPU 模块上的 LED 是处于接下来的那个状态中的信息，按照下列的位模式存储：</div> <div>● 0 是 OFF, 1 是 ON, 2 是闪烁</div> <div><div><div>b15 到 b12</div><div>b11 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div><div>⑧</div><div>⑦</div><div>⑥</div><div>⑤</div><div>④</div><div>③</div><div>②</div><div>①</div></div></div> <div><div><div>①: RUN</div><div>②: 错误</div><div>③: USER</div><div>④: BAT. ALARM</div></div><div><div>⑤: BOOT</div><div>⑥: CARD A (存储卡 A)</div><div>⑦: CARD B (存储卡 B)</div><div>⑧: 空白</div></div></div>	S (状态改变)	新增	QnA
SD202	LED 关闭	LED 的位模式被关闭	<div>● 存储关闭的 LED 位模式 (只有 USER 和 BOOT 还有效)</div> <div>● 在 1 时关闭，在 0 时不关闭</div>	U	新增	QnA
SD203	CPU 的工作状态	CPU 的工作状态	<div>● 远程 I/O 模块的工作状态按照下列格式存储</div> <div><div><div>b15 到 b4</div><div>b3 到 b0</div></div><div><div>to</div><div>b0</div></div></div> <div><div>空白</div><div>①</div></div> <div>① 远程 I/O 模块的工作状态 常 2: STOP</div>	S(一直)	新增	远程
			<div>● CPU 工作状态按照下图指示的格式存储：</div> <div><div><div>b15 到 b12</div><div>b11 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div><div>②</div><div>①</div></div></div> <div><div><div>①: CPU 工作状态</div><div>0: RUN 1: STEPRUN 2: STOP 3: PAUSE</div></div><div><div>②: STOP/PAUSE 原因</div><div>0: RUN/STOP 开关 1: 远程触点 2: 来自 GX Developer 或者串行通信的远程操作 3: 内部程序指令 4: 错误</div></div></div> <div>注意：优先级为最早的最先。</div>	S(每个 END 处理)	D9015 格式改变	○
SD206	设备测试执行类型	0: 没有执行测试 1: 在 X 设备测试过程中 2: 在 Y 设备测试过程中 3: 在 X/Y 设备测试过程中	<div>● 当设备测试模式在 GX Developer 上执行时设置。</div>	S(请求)	新增	远程

特殊寄存器列表 (续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU															
SD207	LED 显示优先级	优先级 1 到 4	<div>● 当错误产生时, LED 显示 (闪烁) 依据于错误号码设置优先级。</div> <div>● 优先级的设置区域如下所示:</div> <div><div>b15 到 b12b11 到 b8b7 到 b4b3 到 b0</div><table><tr><td>SD207</td><td>优先级4</td><td>优先级3</td><td>优先级2</td><td>优先级1</td></tr><tr><td>SD208</td><td>优先级8</td><td>优先级7</td><td>优先级6</td><td>优先级5</td></tr><tr><td>SD209</td><td></td><td></td><td>优先级10</td><td>优先级9</td></tr></table><div>默认值 SD207=4321H SD208=8765H SD209=00A9H</div><div>● 如果“0”被设置, 则无显示。 而且, 即使如果“0”被设置, 关于 CPU 模块操作停止 (包括参数设置) 错误的信息将无条件地由 LED 指示。 参加章节 7.9.5, 关于优先级次序的备注。</div></div> <td>U</td> <td>D9038</td> <td rowspan="3">○</td>	SD207	优先级4	优先级3	优先级2	优先级1	SD208	优先级8	优先级7	优先级6	优先级5	SD209			优先级10	优先级9	U	D9038	○
SD207		优先级4		优先级3	优先级2	优先级1															
SD208		优先级8		优先级7	优先级6	优先级5															
SD209			优先级10	优先级9																	
SD208	优先级 5 到 8	D3039 格式改变																			
SD209	优先级 9 到 10	新增																			
SD210	时钟数据	时钟数据 (年, 月)	<div>● 年 (后两位数字) 和月在 SD210 上如下所示存储成 BCD 代码:</div> <div><div>b15 到 b12b11 到 b8b7 到 b4b3 到 b0</div><div>例子: 七月, 1993 9307H</div></div> <td rowspan="3">S/U (请求)</td> <td>D9025</td> <td rowspan="3">○+Rem</td>	S/U (请求)	D9025	○+Rem															
SD211	时钟数据 (日期, 小时)	<div>● 日期和小时在 SD211 上如下所示存储成 BCD 代码:</div> <div><div>b15 到 b12b11 到 b8b7 到 b4b3 到 b0</div><div>例子: 31号, 10下午 3110H</div></div> <td>D9026</td>	D9026																		
SD212	时钟数据 (分钟, 秒)	<div>● 分钟和秒 (小时之后) 在 SD212 上如下所示存储成 BCD 代码:</div> <div><div>b15 到 b12b11 到 b8b7 到 b4b3 到 b0</div><div>例子: 35分钟, 48秒 3548H</div></div> <td>D9027</td>	D9027																		
SD213	时钟数据	时钟数据 (星期几)	<div>● 年 (两位数字) 和星期几在 SD213 上如下所示存储成 BCD 代码:</div> <div><div>b15 到 b12b11 到 b8b7 到 b4b3 到 b0</div><div>例子: 1993, 星期五 1905H</div><div><div>年的高位数 (0到99)</div><div>星期几</div><table><tr><td>0</td><td>星期日</td></tr><tr><td>1</td><td>星期一</td></tr><tr><td>2</td><td>星期二</td></tr><tr><td>3</td><td>星期三</td></tr><tr><td>4</td><td>星期四</td></tr><tr><td>5</td><td>星期五</td></tr><tr><td>6</td><td>星期六</td></tr></table></div></div> <td rowspan="2">S/U (请求)</td> <td>D9028</td> <td>QCPU 远程</td>	0	星期日	1	星期一	2	星期二	3	星期三	4	星期四	5	星期五	6	星期六	S/U (请求)	D9028	QCPU 远程	
0	星期日																				
1	星期一																				
2	星期二																				
3	星期三																				
4	星期四																				
5	星期五																				
6	星期六																				
<div>● 星期几在 SD213 上如下所示存储成 BCD 代码:</div> <div><div>b15 到 b12b11 到 b8b7 到 b4b3 到 b0</div><div>例子: 星期五 0005H</div><div><div>常设为“0”</div><div>星期几</div><table><tr><td>0</td><td>星期日</td></tr><tr><td>1</td><td>星期一</td></tr><tr><td>2</td><td>星期二</td></tr><tr><td>3</td><td>星期三</td></tr><tr><td>4</td><td>星期四</td></tr><tr><td>5</td><td>星期五</td></tr><tr><td>6</td><td>星期六</td></tr></table></div></div> <td>D9028</td> <td>QnA</td>	0	星期日	1	星期一	2	星期二	3	星期三	4	星期四	5	星期五	6	星期六	D9028	QnA					
0	星期日																				
1	星期一																				
2	星期二																				
3	星期三																				
4	星期四																				
5	星期五																				
6	星期六																				

特殊寄存器列表 (续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU																										
SD220	LED 显示数据	显示指示器数据	● LED 显示 ASCII 数据(16 字符)存储在这里。	S(当被改变时)	新增	○																										
SD221			<table><tr><td>b15 到 b8</td><td>b7 到 b0</td></tr><tr><td>SD220</td><td>从右边数第15个字符</td><td>从右边数第16个字符</td></tr><tr><td>SD221</td><td>从右边数第13个字符</td><td>从右边数第14个字符</td></tr><tr><td>SD222</td><td>从右边数第11个字符</td><td>从右边数第12个字符</td></tr><tr><td>SD223</td><td>从右边数第9个字符</td><td>从右边数第10个字符</td></tr><tr><td>SD224</td><td>从右边数第7个字符</td><td>从右边数第8个字符</td></tr><tr><td>SD225</td><td>从右边数第5个字符</td><td>从右边数第6个字符</td></tr><tr><td>SD226</td><td>从右边数第3个字符</td><td>从右边数第4个字符</td></tr><tr><td>SD227</td><td>从右边数第1个字符</td><td>从右边数第2个字符</td></tr></table>				b15 到 b8	b7 到 b0	SD220	从右边数第15个字符	从右边数第16个字符	SD221	从右边数第13个字符	从右边数第14个字符	SD222	从右边数第11个字符	从右边数第12个字符	SD223	从右边数第9个字符	从右边数第10个字符	SD224	从右边数第7个字符	从右边数第8个字符	SD225	从右边数第5个字符	从右边数第6个字符	SD226	从右边数第3个字符	从右边数第4个字符	SD227	从右边数第1个字符	从右边数第2个字符
b15 到 b8			b7 到 b0																													
SD220			从右边数第15个字符				从右边数第16个字符																									
SD221			从右边数第13个字符				从右边数第14个字符																									
SD222			从右边数第11个字符				从右边数第12个字符																									
SD223			从右边数第9个字符				从右边数第10个字符																									
SD224			从右边数第7个字符				从右边数第8个字符																									
SD225	从右边数第5个字符	从右边数第6个字符																														
SD226	从右边数第3个字符	从右边数第4个字符																														
SD227	从右边数第1个字符	从右边数第2个字符																														
SD222																																
SD223																																
SD224																																
SD225																																
SD226																																
SD227																																
SD240	基本模式	0: 自动模式 1: 详细模式	● 存储此基本模式。	S(初始化)	新增	QCPU 远程																										
SD241	扩展基板的数量	0: 只有基本基板 1 到 7: 扩展基板的数目	● 存储安装的扩展基板的最大数量。	S(初始化)	新增																											
SD242	A/Q 基板的区别	基板类型区别 0: QA * * B 被安装 (A 模式) 1: Q * * B 被安装 (Q 模式)	<table><tr><td>b7</td><td>b2</td><td>b1</td><td>b0</td></tr><tr><td>固定为0</td><td></td><td>到</td><td></td></tr></table> <div><div>→主基板</div><div>→第1扩展基板</div><div>→第2扩展基板</div><div>到</div><div>→第7扩展基板</div></div> <div>当基板没有安装时固定为0</div>	b7	b2		b1	b0	固定为0		到		S(初始化)	新增																		
b7	b2	b1	b0																													
固定为0		到																														
SD243	基板插槽数量	基板插槽数量	<table><tr><td>b15 到 b12</td><td>b11 到 b8</td><td>b7 到 b4</td><td>b3 到 b0</td></tr><tr><td>SD243</td><td>扩展3</td><td>扩展2</td><td>扩展1</td></tr><tr><td>SD244</td><td>扩展7</td><td>扩展6</td><td>扩展5</td></tr><tr><td></td><td></td><td></td><td>主</td></tr></table>	b15 到 b12	b11 到 b8	b7 到 b4	b3 到 b0	SD243	扩展3	扩展2	扩展1	SD244	扩展7	扩展6	扩展5				主	S(初始化)	新增											
b15 到 b12			b11 到 b8	b7 到 b4	b3 到 b0																											
SD243	扩展3	扩展2	扩展1																													
SD244	扩展7	扩展6	扩展5																													
			主																													
SD244	● 如上所示, 每个区域存储了安装的插槽数量。																															
SD250	装入的最大 I/O	装入的最大 I/O 号码	● 当 SM250 从 OFF 变为 ON 时, 装入的模块中最大 I/O 号码的高 2 位加 1 后以 BIN 值存储。	S(请求 END)	新增	○+Rem																										
SD251	用于替换的头 I/O 号码	用于模块替换的头 I/O 号码	● 存储在线状态下, 被拆卸/替换的 I/O 模块的第一个 I/O 号码的高两位。(默认值: 100 _h)	U	D9094	Q2A (S1) Q3A Q4A Q4AR																										
SD253	RS422 传输速率	RS422 传输速率	● 存储 RS422 的传输速率 0: 9600bps 1: 19.2kbps 2: 38.4kbps	S(当被改变时)	新增	QnA																										
SD254	MELSECNET/10 (H) 信息	安装的模块号码	● 表示装配的 MELSECNET/10 模块或者 MELSECNET/H 模块的数量。	S(初始化)	新增	○																										
SD255		I/O 号码	● 装配的 MELSECNET/10 模块或者 MELSECNET/H 模块的 I/O 号码。																													
SD256		网络号码	● 装配的 MELSECNET/10 模块或者 MELSECNET/H 模块的网络号码																													
SD257		组号码	● 装配的 MELSECNET/10 模块或者 MELSECNET/H 模块的组号码																													
SD258		站点号码	● 装配的 MELSECNET/10 模块或者 MELSECNET/H 模块的站点号码																													
SD259		备用信息	● 在备用站点的中, 备用站点的模块号码被存储。(1 到 4)																													
SD260 到 SD264		来自第 2 个模块的信息	● 组态和用于第 2 个模块的相同。																													
SD265 到 SD269		来自第 3 个模块的信息	● 组态和用于第 3 个模块的相同。																													
SD270 到 SD274	来自第 4 个模块的信息	● 组态和用于第 4 个模块的相同。																														

特殊寄存器列表 (续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD280	CC_Link 错误	错误探测状态	<ul style="list-style-type: none">● 当装配的 CC_Link 模块的 Xn0 变为 ON 时，相应站点的位变为 1 (ON)。● 当装配的 CC_Link 模块的 Xn1 或 XnF 中的一个变为 OFF 时，相应站点的位变为 1 (ON)。● 变为 1 (ON)，当装配的 CC_Link 模块和 CPU 模块之间不能通信时。 <p>上面模块的号码 n，是按照头 I/O 号码的顺序。 (然而，没有进行参数设置的模块不被装配。)</p>	S(错误)	新增	QCPU 远程
			<ul style="list-style-type: none">● 当装配的 CC_Link 模块的 Xn0 变为 ON 时，相应站点的位变为 1 (ON)。● 当装配的 CC_Link 模块的 Xn1 或 XnF 中的一个变为 OFF 时，相应站点的位变为 1 (ON)。● 变为 1 (ON)，当装配的 CC_Link 模块和 CPU 模块之间不能通信时。	S(错误)	新增	QnA
SD290	设备分配 (和参数内容相同)	为 X 分配的点数	● 存储当前为 X 设备分配的点数	S(初始化)	新增	○+Rem
SD291		为 Y 分配的点数	● 存储当前为 Y 设备分配的点数			
SD292		为 M 分配的点数	● 存储当前为 M 设备分配的点数			○
SD293		为 L 分配的点数	● 存储当前为 L 设备分配的点数			
SD294		为 B 分配的点数	● 存储当前为 B 设备分配的点数			○+Rem
SD295		为 F 分配的点数	● 存储当前为 F 设备分配的点数			○
SD296		为 SB 分配的点数	● 存储当前为 SB 设备分配的点数			○+Rem
SD297		为 V 分配的点数	● 存储当前为 V 设备分配的点数			○
SD298		为 S 分配的点数	● 存储当前为 S 设备分配的点数			
SD299		为 T 分配的点数	● 存储当前为 T 设备分配的点数			
SD300		为 ST 分配的点数	● 存储当前为 ST 设备分配的点数			
SD301		为 C 分配的点数	● 存储当前为 C 设备分配的点数			

特殊寄存器列表 (续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU	
SD302	设备分配 (和参数内容相同)	为 D 分配的点数	● 存储当前为 D 设备分配的点数	S(初始化)	新增	○+Rem	
SD303		为 W 分配的点数	● 存储当前为 W 设备分配的点数				
SD304		为 SW 分配的	● 存储当前为 SW 设备分配的点数				
SD315	保留给通信处理的时间	保留给通信处理的时间	保留指定的时间给与 GX Developer 或其它单元通信的时间。 指定的值越大, 和其它设备(GX Developer, 串行通信单元)通信的响应时间就越短。 设置范围: 1 到 100ms 如果指定值超出了上述范围, 则被假定为没有设置。 扫描时间增加指定的时间。	END 处理	新增	QCPU	
SD340	以太网信息	安装的模块数量	● 表示装配的以太网模块的数量。	S(初始化)	新增	QCPU 远程	
SD341		第 1 个模块的信息	I/O 号码				● 装配的以太网模块的 I/O 号码
SD342			网络号码				● 装配的以太网模块的网络号码
SD343			组号码				● 装配的以太网模块的组号码
SD344			站点号码				● 装配的以太网模块的站点号码
SD345 到 SD346			空白				● 空白 (对 QCPU, 第 1 个模块的以太网 IP 地址存储在缓冲存储区内。)
SD347		空白	● 空白 (对 QCPU, 第 1 个模块的以太网错误代码由错误 RD 指令读取。)				
SD348 到 SD354		第 2 个模块的信息	● 组态和第 1 个模块的相同。	S(初始化)	新增		
SD355 到 SD361		第 3 个模块的信息	● 组态和第 1 个模块的相同。				
SD362 到 SD368		第 4 个模块的信息	● 组态和第 1 个模块的相同。				
SD340	以太网信息	安装的模块数量	● 表示装配的以太网模块的数量。	S(初始化)	新增	QnA	
SD341		第 1 个模块的信息	I/O 号码				● 装配的以太网模块的 I/O 号码
SD342			网络号码				● 装配的以太网模块的网络号码
SD343			组号码				● 装配的以太网模块的组号码
SD344			站点号码				● 装配的以太网模块的站点号码
SD345 到 SD346			IP 地址				● 装配的以太网模块的 IP 地址
SD347		错误代码	● 装配的以太网模块的错误代码				
SD348 到 SD354		第 2 个模块的信息	● 组态和第 1 个模块的相同。	S(初始化)	新增		
SD355 到 SD361		第 3 个模块的信息	● 组态和第 1 个模块的相同。				
SD362 到 SD368		第 4 个模块的信息	● 组态和第 1 个模块的相同。				
SD380	以太网指令接收状态	第 1 个模块的指令接收状态	<div><div>b15 到 b8 b7 b6 b5 b4 b3 b2 b1 b0</div><div>0 到 0 <input type="checkbox"/><input type="checkbox"/><input type="checkbox"/><input type="checkbox"/><input type="checkbox"/><input type="checkbox"/><input type="checkbox"/><input type="checkbox"/></div><div>未使用</div><div><div>通道1的指令接收状态</div><div>通道2的指令接收状态</div><div>通道3的指令接收状态</div><div>通道4的指令接收状态</div><div>通道5的指令接收状态</div><div>通道6的指令接收状态</div><div>通道7的指令接收状态</div><div>通道8的指令接收状态</div></div><div>ON: 已接收(通道被使用) OFF: 未接收(通道未被使用)</div></div>	S(初始化)	新增		

特殊寄存器列表(续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD381	以太网指令接收状态	第 2 个模块的信息	● 组态和第 1 个模块的相同。	S(初始化)	新增	QnA
SD382		第 3 个模块的信息	● 组态和第 1 个模块的相同。			
SD383		第 4 个模块的信息	● 组态和第 1 个模块的相同。			
SD392	软件版本	内部系统软件版本	<p>● 以 ASCII 码存储内部系统软件版本。</p> <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 5px;">高字节</div> <div style="border: 1px solid black; padding: 2px; margin-right: 5px;">低字节</div> <div style="text-align: center; margin-left: 5px;">↑</div> </div> <p>低字节中的数据是不确定的。 软件版本存储在高字节位置。 对于版本“A”，举例来说，“41H”将被存储</p> <p>注意：内部系统软件版本可能会和包装上打印的版本符号指示的版本号不同。</p>	S(初始化)	D9060	
SD395	多 PLC 号码	多 PLC 号码	<p>● 在一个多 PLC 系统的组态中，主机 CPU 的 PLC 号码被存储。 1 号 PLC: 1, 2 号 PLC: 2, 3 号 PLC: 3, 4 号 PLC: 4</p>	S(初始化)	新增	QCPU 功能版本 B 或更晚

(3) 系统时钟/计数器

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD412	1 秒计数器	1 秒单位内的计数量	<p>● 可编程控制器的 CPU 模块变为 RUN 后，每秒增加 1。</p> <p>● 计数从 0 到 32767 到 32768 到 0 重复。</p>	S (状态改变)	D9022	○
SD414	2n 秒时钟设置	2n 秒时钟单位	<p>● 存储 2n 秒时钟的值 n(默认为 30)</p> <p>● 设置可以在 1 和 32767 之间进行</p>	U	新增	
SD415	2n 毫秒时钟设置	2n 毫秒时钟单位	<p>● 存储 2n 毫秒时钟的值 n(默认为 30)</p> <p>● 设置可以在 1 和 32767 之间进行</p>	U	新增	QCPU
SD420	扫描计数器	每个扫描周期内的计数量	<p>● 在 CPU 模块被设为 RUN 后，每个扫描执行增加 1。*</p> <p>● 计数从 0 到 32767 到 32768 到 0 重复</p>	S(每个 END 处理)	新增	○
SD430	低速扫描计数器	每个扫描周期内的计数量	<p>● 在 CPU 模块被设为 RUN 后，每个扫描执行增加 1。</p> <p>● 计数从 0 到 32767 到 32768 到 0 重复</p> <p>● 只能用于低速执行类型的程序</p>	S(每个 END 处理)	新增	

*：在初始化执行类型程序中不被扫描周期计算。

特殊寄存器列表

(4) 扫描信息

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD500	执行程序号码	执行中的程序号码	● 当前被执行程序的程序号码被以 BIN 值存储。	S (状态改变)	新增	
SD510	低速执行类型程序号码	低速执行类型程序号码	● 当前被执行低速执行类型程序的程序号码被以 BIN 值存储。 ● 只有在 SM510 为 ON 时被使能。	S(每个 END 处理)	新增	
SD520	当前扫描时间	当前扫描时间 (以 1ms 为单位)	● 存储当前扫描时间(以 1 ms 为单位) 范围从 0 到 65535	S(每个 END 处理)	D9017 格式改变	
SD521		当前扫描时间 (以 100μs 为单位)	● 存储当前扫描时间(以 100μs 为单位) 范围从 00000 到 900 (例子) 当前扫描时间 23.6ms 被存储成以下形式: D520=23 D521=600	S(每个 END 处理)	新增	
SD522	初始扫描时间	初始扫描时间 (以 1ms 为单位)	● 将初始化执行类型程序的扫描时间存储到 SD522 和 SD523。(测量是以 100μs 为单位) SD522: 存储 ms 位置。(存储范围: 0 到 65535) SD523: 存储 μs 位置。(存储范围: 0 到 900)	S(第一个 END 处理)	新增	
SD523		初始扫描时间 (以 100μs 为单位)				
SD524	最小扫描时间	最小扫描时间 (以 1ms 为单位)	● 将除了初始化执行类型程序外, 其它程序扫描时间的最小值存储到 SD524 和 SD525。(测量是以 100μs 为单位) SD524: 存储 ms 位置。(存储范围: 0 到 65535) SD525: 存储 μs 位置。(存储范围: 0 到 900)	S(每个 END 处理)	D9018 格式改变	
SD525		最小扫描时间 (以 100μs 为单位)		S(每个 END 处理)	新增	
SD526	最大扫描时间	最大扫描时间 (以 1ms 为单位)	● 将除了初始化执行类型程序外, 其它程序扫描时间的最大值存储到 SD526 和 SD527。(测量是以 100μs 为单位.) SD526: 存储 ms 位置。(存储范围: 0 到 65535) SD527: 存储 μs 位置。(存储范围: 0 到 900)	S(每个 END 处理)	D9019 格式改变	○
SD527		最大扫描时间 (以 100μs 为单位)			新增	
SD528	低速执行类型程序的当前扫描时间	当前扫描时间 (以 1ms 为单位)	● 将低速执行类型程序的当前扫描时间存储到 SD528 和 SD529。(测量是以 100μs 为单位) SD528: 存储 ms 位置。(存储范围: 0 到 65535) SD529: 存储 μs 位置。(存储范围: 0 到 900)	S(每个 END 处理)	新增	
SD529		当前扫描时间 (以 100μs 为单位)				
SD532	低速执行类型程序的最小扫描时间	最小扫描时间 (以 1ms 为单位)	● 将低速执行类型程序扫描时间的最小值存储到 SD532 和 SD533。(测量是以 100ms 为单位) SD532: 存储 ms 位置。(存储范围: 0 到 65535) SD533: 存储 μs 位置。(存储范围: 0 到 900)	S(每个 END 处理)	新增	
SD533		最大扫描时间 (以 100μs 为单位)				
SD534	低速执行类型程序的最大扫描时间	最大扫描时间 (以 1ms 为单位)	● 将低速执行类型程序除了第一个扫描周期外, 其它扫描时间的最大值存储到 SD534 和 SD535。(测量是以 100μs 为单位) SD534: 存储 ms 位置。(存储范围: 0 到 65535) SD535: 存储 μs 位置。(存储范围: 0 到 900)	S(每个 END 处理)	新增	
SD535		最大扫描时间 (以 100μs 为单位)				
SD540	END 处理时间	END 处理时间 (以 1ms 为单位)	● 将扫描周期执行类型程序从一个扫描周期结束到下一个扫描周期开始的时间存储到 SD540 和 SD541。(测量是以 100μs 为单位) SD540: 存储 ms 位置。(存储范围: 0 到 65535) SD541: 存储 μs 位置。(存储范围: 0 到 900)	S(每个 END 处理)	新增	
SD541		END 处理时间 (以 100μs 为单位)				
SD542	固定扫描等待时间	固定扫描等待时间 (以 1ms 为单位)	● 将固定扫描设置的等待时间存储到 SD542 和 SD543。(测量是以 100μs 为单位) SD542: 存储 ms 位置。(存储范围: 0 到 65535) SD543: 存储 μs 位置。(存储范围: 0 到 900)	S(第一个 END 处理)	新增	
SD543		固定扫描等待时间 (以 100μs 为单位)				

特殊寄存器列表 (续表)

号码	名称	含义	解释	设置者 (当被设置时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD544	低速执行类型程序的累计执行时间	低速执行类型程序的累计执行时间 (以 1ms 为单位)	<ul style="list-style-type: none">将低速执行类型程序的累计执行时间存储到 SD544 和 SD545。(测量是以 100μs 为单位)SD544: 存储 ms 位置(存储范围: 0 到 65535)SD545: 存储 μs 位置 (存储范围: 0 到 900)在一个低速扫描周期结束后, 清为 0。	S(每个 END 处理)	新增	○
SD545		低速执行类型程序的累计执行时间 (以 100μs 为单位)				
SD546	低速执行类型程序的执行时间	低速执行类型程序的执行时间 (以 1ms 为单位)	<ul style="list-style-type: none">将低速执行类型程序一个扫描周期内的执行时间存储到 SD546 和 SD547。(测量是以 100μs 为单位)SD546: 存储 ms 位置(存储范围: 0 到 65535)SD547: 存储 μs 位置 (存储范围: 0 到 900)在每个扫描周期存储。	S(每个 END 处理)	新增	
SD547		低速执行类型程序的执行时间 (以 100μs 为单位)				
SD548	扫描执行类型程序执行时间	扫描执行类型程序执行时间 (以 1ms 为单位)	<ul style="list-style-type: none">将扫描执行类型程序一个扫描周期内的执行时间存储到 SD548 和 SD549。(测量是以 100μs 为单位)SD548: 存储 ms 位置 (存储范围: 0 到 65535)SD549: 存储 μs 位置 (存储范围: 0 到 900)在每个扫描周期存储。	S(每个 END 处理)	新增	
SD549		扫描执行类型程序执行时间 (以 100μs 为单位)				
SD550	服务间隔测量模块	单元/模块号码	<ul style="list-style-type: none">为测量服务间隔的模块设置 I/O 号码	U	新增	○+Rem
SD551	服务间隔时间	模块服务间隔 (以 1 ms 为单位)	<ul style="list-style-type: none">当 SM551 变为 ON 时, 将用于在 SD550 中指定模块的服务间隔存储到 SD551 和 SD552(测量是以 100ms 为单位)SD551: 存储 ms 位置(存储范围: 0 到 65535)SD552: 存储 μs 位置(存储范围: 0 到 900)	S(请求)	新增	
SD552		模块服务间隔 (以 100μs 为单位)				

特殊寄存器列表

(5) 存储卡

号码	名称	含义	解释	设定者 (当被设定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD600	存储卡模块	存储卡模块	<div>● 表示安装的存储卡模块</div> <div><div><div>b15 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div><div>0</div><div></div><div></div></div><div><div>驱动器1 (RAM) 模型</div><div>0: 不存在 1: SRAM</div></div><div><div>驱动器2 (ROM) 模型</div><div>0: 不存在 (1: SRAM) 2: ATA闪存 3: 闪存ROM</div></div></div>	S(初始化和卡拆除)	新增	QCPU
	存储卡 A 模块	存储卡 A 模块	<div>● 表示安装的存储卡 A 模块</div> <div><div><div>b15 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div><div>0</div><div></div><div></div></div><div><div>驱动器1 (RAM) 模型</div><div>0: 不存在 1: SRAM</div></div><div><div>驱动器2 (ROM) 模型</div><div>0: 不存在 2: EEPROM 3: 闪存ROM</div></div></div>	S(初始化和卡拆除)	新增	QnA
SD602	驱动器 1 (RAM) 容量	驱动器 1 容量	<div>● 驱动器 1 容量以 1k 字节为单位存储</div>	S(初始化和卡拆除)	新增	QCPU
SD603	驱动器 2 (ROM) 容量	驱动器 2 容量	<div>● 驱动器 2 容量以 1k 字节为单位存储</div>	S(初始化和卡拆除)	新增	QnA
				S(初始化和卡拆除)	新增	QCPU
SD604	存储卡使用条件	存储卡使用条件	<div>● 存储卡 (A) 的使用条件以位样式存储 (在使用时为 0N)</div> <div>● 这些位样式的意义如下所示:</div> <div><div><div>b0: 导入操作 (QBT) b1: 参数 (QPA) b2: 设备注释 (QCD) b3: 设备初始值 (QDI) b4: 文件寄存器 R (QDR) b5: 跟踪 (QTS) b6: 未使用 b7: 未使用</div><div>b8: 未使用 — b9: CPU 故障历史 (QFD) b10: 未使用 b11: 本地设备 (QDL) b12: 未使用 b13: 未使用 b14: 未使用 b15: 未使用</div></div></div>	S(状态改变)	新增	QCPU
	存储卡 A 使用条件	存储卡 A 使用条件	<div>● 存储卡 A 的使用条件以位样式存储 (在使用时为 0N)</div> <div>● 这些位样式的意义如下所示:</div> <div><div><div>b0: 导入操作 (QBT) b1: 参数 (QPA) b2: 设备注释 (QCD) b3: 设备初始值 (QDI) b4: 文件寄存器 R (QDR) b5: 采样跟踪 (QTS) b6: 状态闭锁 (QTL) b7: 程序跟踪 (QTP)</div><div>b8: 仿真数据 (QDS) b9: CPU 故障历史 (QFD) b10: SFC 跟踪 (QTR) b11: 本地设备 (QDL) b12: 未使用 b13: 未使用 b14: 未使用 b15: 未使用</div></div></div>	S(状态改变)	新增	QnA
SD620	驱动器 3/4 模块	驱动器 3/4 模块	<div>● 表示驱动器 3/4 模块。</div> <div><div><div>b15 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div><div>0</div><div></div><div></div></div><div><div>驱动器3 (标准RAM)</div><div>固定在“1”。</div></div><div><div>驱动器4 (标准ROM)</div><div>固定在“3”。</div></div></div>	S(初始化)	新增	QCPU

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定 时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD620	存储卡 B 模块	存储卡 B 模块	<div><div>● 表示安装的存储卡 B 模块</div><div><div><div>b15 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div><div>0</div></div></div><div><div>驱动器1 (RAM) 模型</div><div>0: 不存在 1: SRAM</div></div><div><div>驱动器2 (ROM) 模型</div><div>0: 不存在 2: E²PROM 3: 闪存ROM</div></div></div>	S (初始化)	新增	Q2A (S1) Q3A Q4A Q4AR
SD622	驱动器 3 (RAM) 容量	驱动器 3 容量	● 驱动器 3 容量以 1k 字节为单位存储。	S (初始化)	新增	QCPU
			● 驱动器 3 容量以 1k 字节为单位存储。	S (初始化)	新增	Q2A (S1) Q3A Q4A Q4AR
SD623	驱动器 4 (ROM) 容量	驱动器 4 容量	● 驱动器 4 容量以 1k 字节为单位存储。	S (初始化)	新增	QCPU
			● 驱动器 4 容量以 1k 字节为单位存储。	S (初始化)	新增	Q2A (S1) Q3A Q4A Q4AR
SD624	驱动器 3/4 使用 条件	驱动器 3/4 使用 条件	<div><div>● 驱动器 3/4 的使用条件以位样式存储。 (在使用时为 ON)</div><div>● 这些位样式的意义如下所示:</div><div><div><div>b0: 导入操作 (QBT)</div><div>b1: 参数 (QPA)</div><div>b2: 设备注释 (QCD)</div><div>b3: 设备初始化值 (QDI)</div><div>b4: 文件寄存器 (QDR)</div><div>b5: 跟踪 (QTS)</div><div>b6: 未使用</div><div>b7: 未使用</div></div><div><div>b8: 未使用</div><div>b9: CPU 故障历史 (QFD)</div><div>b10: 未使用</div><div>b11: 本地设备 (QDL)</div><div>b12: 未使用</div><div>b13: 未使用</div><div>b14: 未使用</div><div>b15: 未使用</div></div></div></div>	S (状态 改变)	新增	QCPU
	存储卡 B 使用 条件	存储卡 B 使用 条件	<div><div>● 存储卡 B 使用条件以位样式存储。 (在使用时为 ON)</div><div>● 这些位样式的意义如下所示:</div><div><div><div>b0: 导入操作 (QBT)</div><div>b1: 参数 (QPA)</div><div>b2: 设备注释 (QCD)</div><div>b3: 设备初始化值 (QDI)</div><div>b4: 文件寄存器 (QDR)</div><div>b5: 采样跟踪 (QTS)</div><div>b6: 状态闭锁 (QTL)</div><div>b7: 程序跟踪 (QTP)</div></div><div><div>b8: 仿真数据 (QDS)</div><div>b9: CPU 故障历史 (QFD)</div><div>b10: SFC 跟踪 (QTR)</div><div>b11: 本地设备 (QDL)</div><div>b12: 未使用</div><div>b13: 未使用</div><div>b14: 未使用</div><div>b15: 未使用</div></div></div></div>	S (状态 改变)	新增	Q2A (S1) Q3A Q4A Q4AR
SD640	文件寄存器驱动器	驱动器号码:	● 存储这被文件寄存器使用的驱动器号码	S (初始化)	新增	○
SD641	文件寄存器容量	文件寄存器容量	● 以 ASCII 码存储在参数上选择的或被 QDRSET 指令使用的文件寄存器文件名称 (带扩展的)。 <div><div><div>b15 到 b8</div><div>b7 到 b0</div></div><div><div>SD641 第二个字符</div><div>第一个字符</div></div><div><div>SD642 第四个字符</div><div>第三个字符</div></div><div><div>SD643 第六个字符</div><div>第五个字符</div></div><div><div>SD644 第八个字符</div><div>第七个字符</div></div><div><div>SD645 扩展的第一个字符</div><div>2E# (.)</div></div><div><div>SD646 扩展的第三个字符</div><div>扩展的第二个字符</div></div></div>	S (初始化)	新增	
SD642						
SD643						
SD644						
SD645						
SD646						
SD647	文件寄存器容量	文件寄存器容量	● 以 1k 字为单位存储当前选择的文件寄存器的数据容量。	S (状态 改变)	新增	
SD648	文件寄存器块 号码	文件寄存器块 号码	● 存储当前选择的文件寄存器块号码。	S (状态 改变)	D9035	
SD650	注释驱动器	注释驱动器号 码	● 存储在参数上选择的或被 QCDSET 指令使用的注释驱动器的号码。	S (状态 改变)	新增	

特殊寄存器列表 (续表)

号码	名称	含义	解释	解释 (当被设 定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD651	注释文件名称	注释文件名称	● 将在参数上选择的或由 QCDSET 指令使用的注释文件名称(带扩展)存储成 ASCII 码。	S(状态 改变)	新增	○
SD652						
SD653						
SD654						
SD655						
SD656						
SD660	导入操作指定 文件	导入指定文件 驱动器号码	● 存储导入指定文件(*.QBT)被保存的驱动器的号码。	S(初始化)	新增	
SD661		导入指定文件 的文件名称	● 存储导入指定文件(*.QBT)的文件名称。	S(初始化)	新增	
SD662						
SD663						
SD664						
SD665						
SD666						

(6) 和寄存器有关的指令

号码	名称	含义	解释	设定者 (当被设定 时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU																				
SD705	掩码模式	掩码模式	● 在块操作中, 将 SM705 变为 ON, 以便能使用存储在 SD705(或者在 SD705 和 SD706 内, 如果使用的是双字数据)中的掩码样式按照掩码值来操作块中的所有数据。	U	新增	○																				
SD706																										
SD714	空白通讯请求 登记区域的数量	0 到 32	● 存储用于连接到 MELSECNET/MINI-S3 的远程终端模块的通讯请求区域内, 空白块的数量。	S(在执行 过程中)	M9081	QnA																				
SD715	IMASK 指令掩 码模式	掩码模式	● 使用 IMASK 指定掩码的模式按照下列样式存储: <table><tr><td></td><td>b15</td><td>到</td><td>b1</td><td>b0</td></tr><tr><td>SD715</td><td>115</td><td></td><td>11</td><td>10</td></tr><tr><td>SD716</td><td>131</td><td>到</td><td>117</td><td>116</td></tr><tr><td>SD717</td><td>147</td><td>到</td><td>133</td><td>132</td></tr></table>		b15	到	b1	b0	SD715	115		11	10	SD716	131	到	117	116	SD717	147	到	133	132	S(在执行 过程中)	新增	○
				b15	到	b1	b0																			
SD715				115		11	10																			
SD716	131	到	117	116																						
SD717	147	到	133	132																						
SD716																										
SD717																										
SD718	累加器	累加器	● 用作用于 A-系列程序中的累加器的替代。	S/U	新增																					
SD719																										
SD720	用于 PLOAD 指 令的程序号码 指定	用于 PLOAD 指 令的程序号码 指定	● 当被指定时存储被 PLOAD 指令装载的程序的程序号码。 ● 指定范围: 1 到 124	U	新增	QCPU																				
SD730	用于 CC-Link 通讯请求的的 空白登记区域 的数量	0 到 32	● 存储用于和连接到 A(1S)J61QBT61 的智能设备站点进行通讯的请求的空白登记区域数量。	S(在执行 过程中)	新增	QnA																				
SD736	PKEY 输入	PKEY 输入	● 特殊寄存器, 能临时保存使用 PKEY 指令输入的键盘数据。	S(在执行 过程中)	新增	QnA																				

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定 定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD738	消息存储	消息存储	● 存储由 MSG 指令指定的消息	S(在执行 过程中)	新增	○
SD739						
SD740						
SD741						
SD742						
SD743						
SD744						
SD745						
SD746						
SD747						
SD748						
SD749						
SD750						
SD751						
SD752						
SD753						
SD754						
SD755						
SD756						
SD757						
SD758						
SD759						
SD760						
SD761						
SD762						
SD763						
SD764						
SD765						
SD766						
SD767						
SD768						
SD769						
SD774 TO SD775	PID 界限设定 (用于完整微分)	0: 界限设定 1: 界限未设定	● 按照如下所示指定每个 PID 回路界限: <div><div><div>b15到b1</div><div>SD774回路16到回路2回路1</div><div>SD775回路32到回路18回路17</div></div></div>	U	新增	QCPU
SD778	当 COM 指令被 执行时的刷新 处理选择	b0 到 b14: 0: 不刷新 1: 刷新 b15 位 0: 执行一般数 据处理 1: 不执行一般 数据处理	● 选择当 COM 指令被执行时数据是否被刷新。 ● 选择当 COM 指令被执行时数据是否被刷新。 <div><div><div>b15b14到b5b4b3b2b1b0</div><div>SD7781/001/01/01/01/0</div><div>I/O 刷新</div><div>CC-Link刷新</div><div>MELSECNET/H刷新</div><div>智能功能模块的自动刷新</div><div>CPU共享存储器的自动刷新</div><div>执行 / 不执行一般数据处理</div></div></div>	U	新增	QCPU 系列号 04012 或更大

特殊寄存器列表

(7) 调试

号码	号码	含义	解释	设定者 (当被设定 时)	相应的 ACPU D9 <table><tr><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr></table>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	相应的 CPU																																																						
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																													
SD780	CC-Link 专用 指令同时执行 剩下的数目	0 到 32	● 存储 CC-Link 专用指令同时执行剩下的数目。	U	新增	QnA																																																									
SD781 TO SD793	IMASK 指令的 掩码模式	掩码模式	● 按照如下所示存储 IMASK 指令的掩码模式： <table><tr><td></td><td>b15</td><td></td><td>b1</td><td>b0</td></tr><tr><td>SD781</td><td>163</td><td>到</td><td>149</td><td>148</td></tr><tr><td>SD782</td><td>179</td><td>到</td><td>165</td><td>164</td></tr><tr><td></td><td colspan="4">到</td></tr><tr><td>SD793</td><td>1255</td><td>到</td><td>1241</td><td>1240</td></tr></table>		b15		b1	b0	SD781	163	到	149	148	SD782	179	到	165	164		到				SD793	1255	到	1241	1240	S(在执行 过程中)	新增	QCPU																																
	b15		b1	b0																																																											
SD781	163	到	149	148																																																											
SD782	179	到	165	164																																																											
	到																																																														
SD793	1255	到	1241	1240																																																											
SD794 TO SD795	PID 界限设定 (用于完整微 分)	0: 界限设定 1: 界限未 设定	● 按照如下所示指定每个 PID 回路界限： <table><tr><td></td><td>b15</td><td></td><td>到</td><td>b1</td><td>b0</td></tr><tr><td>SD794</td><td>回路16</td><td></td><td>到</td><td>回路2</td><td>回路1</td></tr><tr><td>SD795</td><td>回路32</td><td></td><td>到</td><td>回路18</td><td>回路17</td></tr></table>		b15		到	b1	b0	SD794	回路16		到	回路2	回路1	SD795	回路32		到	回路18	回路17	U	新增	QCPU 系列号 05032 或更大																																							
	b15		到	b1	b0																																																										
SD794	回路16		到	回路2	回路1																																																										
SD795	回路32		到	回路18	回路17																																																										
SD806	状态闭锁文件 名称	状态闭锁文件 名称	● 当状态闭锁被执行时，立即以 ASCII 码形式从数据点存储文件名称 （带扩展）。 <table><tr><td></td><td>b15</td><td></td><td>到</td><td>b8</td><td>b7</td><td></td><td>到</td><td>b0</td></tr><tr><td>SD806</td><td colspan="4">第二个字符</td><td colspan="3">第一个字符</td></tr><tr><td>SD807</td><td colspan="4">第四个字符</td><td colspan="3">第三个字符</td></tr><tr><td>SD808</td><td colspan="4">第六个字符</td><td colspan="3">第五个字符</td></tr><tr><td>SD809</td><td colspan="4">第八个字符</td><td colspan="3">第七个字符</td></tr><tr><td>SD810</td><td colspan="4">扩展的第一个字符</td><td colspan="3">2EH (.)</td></tr><tr><td>SD811</td><td colspan="4">扩展的第三个字符</td><td colspan="3">扩展的第二个字符</td></tr></table>		b15		到	b8	b7		到	b0	SD806	第二个字符				第一个字符			SD807	第四个字符				第三个字符			SD808	第六个字符				第五个字符			SD809	第八个字符				第七个字符			SD810	扩展的第一个字符				2EH (.)			SD811	扩展的第三个字符				扩展的第二个字符			S(在执行 过程中)	新增	QnA
			b15		到	b8	b7		到	b0																																																					
SD806			第二个字符				第一个字符																																																								
SD807			第四个字符				第三个字符																																																								
SD808			第六个字符				第五个字符																																																								
SD809			第八个字符				第七个字符																																																								
SD810	扩展的第一个字符				2EH (.)																																																										
SD811	扩展的第三个字符				扩展的第二个字符																																																										
SD807																																																															
SD808																																																															
SD809																																																															
SD810																																																															
SD811																																																															
SD812	状态闭锁步	状态闭锁步	● 当状态闭锁被执行时，立即从数据点存储步号码。 <table><tr><td>SD812</td><td>模式*</td></tr><tr><td>SD813</td><td>块号码</td></tr><tr><td>SD814</td><td>步号码/转换条件号码</td></tr><tr><td>SD815</td><td>顺序步号码(L)</td></tr><tr><td>SD816</td><td>顺序步号码(H)</td></tr></table> * 模式数据的内容 <table><tr><td>15</td><td>14</td><td>到</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>← (位号码)</td></tr><tr><td>0</td><td>0</td><td>到</td><td>0</td><td>0</td><td>*</td><td>*</td><td>*</td><td></td></tr></table> (未使用)	SD812	模式*	SD813	块号码	SD814	步号码/转换条件号码	SD815	顺序步号码(L)	SD816	顺序步号码(H)	15	14	到	4	3	2	1	0	← (位号码)	0	0	到	0	0	*	*	*																																	
SD812			模式*																																																												
SD813			块号码																																																												
SD814			步号码/转换条件号码																																																												
SD815			顺序步号码(L)																																																												
SD816	顺序步号码(H)																																																														
15	14	到	4	3	2	1	0	← (位号码)																																																							
0	0	到	0	0	*	*	*																																																								

特殊寄存器列表

(8) 闭锁区域

号码	名称	含义	解释	设定者 (当被设定 定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU	
SD900	电源发生中断的驱动器	在电源掉电过程中，访问文件驱动器号码	● 在电源掉电过程中，如果文件正被访问，存储驱动器号码。	S(状态改变)	新增		
SD901	在电源掉电过程中，活动文件名称	在电源掉电过程中，访问文件名称	● 在电源掉电过程中，如果文件正被访问，以 ASCII 码存储文件名称(带扩展)。	S(状态改变)	新增		
SD902							
SD903							
SD904							
SD905							
SD906							
SD910	RKEY 输入	RKEY 输入	● 按照 PU 键码输入顺序存储。	S(在执行过程中)	新增		
SD911							
SD912							
SD913							
SD914							
SD915							
SD916							
SD917							
SD918							
SD919							
SD920							
SD921							
SD922							
SD923							
SD924							
SD925							

- (9) A 到 Q/QnA 转换的对应
- 在 A 系列到 Q/QnA 系列的转换后，ACPU 的特殊寄存器 D9000 到 D9255 对应特殊寄存器 SD1000 到 SD1255。
- 这些寄存器都由系统设定，用户不能使用它们去设置程序数据。
- 需要使用这些寄存器设置数据的用户应该修改用于 Q/QnA 的特殊寄存器。
- 然而，在转换之前用户只能在特殊寄存器 D9200 到 D9255 上设置数据，在转换之后，用户就也可以在寄存器 1200 到 1255 上设置数据了。
- 关于 ACPU 特殊寄存器的内容的详细信息，参见每个 CPU 的用户手册，以及 MELSECNET 和 MELSECNET/B 数据链接系统参考手册。

备注

“用于修改的特殊寄存器”栏的附加解释。

- ① 用于指定用于修改的特殊寄存器的软元件号，修改它以适应用于 QCPU/QnACPU 的特殊寄存器。
- ② 用于 ☐ 被指定的软元件号，转换后的特殊寄存器能被使用。
- ③ 对于 ☒ 被指定的软元件号，对 QCPU/QnACPU 不起作用。

特殊寄存器列表

ACPU 特殊转换	转换后特 殊寄存器	用于修改 特殊寄存器	名称	含义	详细内容	相应的 CPU																																							
D9000	SD1000	—	保险丝烧坏	保险丝烧坏的模块 的数量	<ul style="list-style-type: none">当保险丝烧坏的模块被探测到时，探测到的模块的最低号码的第一个 I/O 号码被以十六进制数据格式存储。 (例子: 当 Y50 到 6F 输出模块保险丝被烧坏, “50” 被以十六进制存储)通过使用外部设备来监视此号码，执行以十六进制给出的监视操作。 (当 SD1100 到 SD1107 中所有内容被复位到 0 时，寄存器中的内容被清除)远程 I / O 站点的输出模块也被执行保险丝烧坏检查。	○																																							
D9001	SD1001	—	保险丝烧坏	保险丝烧坏的模块 的数量	<ul style="list-style-type: none">当发生保险丝烧坏时，存储和设置开关号码或者基板插槽号码相对应的模块号码。 <table><tr><th colspan="2">用于 A0J2 的 I/O 模块</th><th colspan="2">扩展基板单元</th></tr><tr><th>设置开关</th><th>存储的数据</th><th>基板单元插槽号码</th><th>存储的数据</th></tr><tr><td>0</td><td>1</td><td>0</td><td>5</td></tr><tr><td>1</td><td>2</td><td>1</td><td>6</td></tr><tr><td>2</td><td>3</td><td>2</td><td>7</td></tr><tr><td>3</td><td>4</td><td>3</td><td>8</td></tr><tr><td>4</td><td>5</td><td></td><td></td></tr><tr><td>5</td><td>6</td><td></td><td></td></tr><tr><td>6</td><td>7</td><td></td><td></td></tr><tr><td>7</td><td>8</td><td></td><td></td></tr></table>		用于 A0J2 的 I/O 模块		扩展基板单元		设置开关	存储的数据	基板单元插槽号码	存储的数据	0	1	0	5	1	2	1	6	2	3	2	7	3	4	3	8	4	5			5	6			6	7			7	8	
用于 A0J2 的 I/O 模块		扩展基板单元																																											
设置开关	存储的数据	基板单元插槽号码	存储的数据																																										
0	1	0	5																																										
1	2	1	6																																										
2	3	2	7																																										
3	4	3	8																																										
4	5																																												
5	6																																												
6	7																																												
7	8																																												

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后特 殊寄存器	用于修改 特殊寄存器	名称	含义	详细内容	相应的 CPU
D9002	SD1002	—	I/O 模块验证错误	I/O 模块验证错误 模块号码	<ul style="list-style-type: none"> 在电源上电时如果其数据和输入的数据不同的 I / O 模块被检测到, 被检测到的单元中最低号码单元的 第一个 I / O 号码被以十六进制数据格式存储。(存储的方法和 SD1000 相同) 通过使用外部设备监视此号码, 执行十六进制数据中给出的监视操作。 (当 SD1116 到 SD1123 中所有内容被复位到 0 时, 寄存器中的内容被清除) 远程 I/O 终端的模块也执行 I/O 模块验证检查。 	○
D9004	SD1004	—	MINI 链接错误	存储在参数上设置的 设置状态(模块 1 到 8)	<div> <div> b15 到 b8 </div> <div> 8 7 6 5 4 3 2 1 </div> </div> <div> <div> b7 到 b0 </div> <div> 8 7 6 5 4 3 2 1 </div> </div> <div> 和故障相对应的位AJ71PT32 (S3) 变为ON </div> <div> 如下所示, 在信号变为ON时, 和AJ71PT32 (S3) 的信号相对应的位被变为ON • 硬件错误 (X0/X20) • MINI (S3) 链接错误检测 (X6/X26) • MINI (S3) 链接通信错误 (X7/X27) </div>	QnA
D9005	SD1005	—	AC DOWN 计数器	AC DOWN 的次数	当使用 AC 电源供电模块时, 每当发生 20ms 内的瞬时电源故障时就加 1。 (此值以 BIN 码存储) 当电源从 OFF 切换到 ON 时计数器被复位。	○
					当使用 DC 电源供电模块时, 每当发生 10ms 内的瞬时电源故障时就加 1。 (此值以 BIN 码存储) 当电源从 OFF 切换到 ON 时计数器被复位。	QnA
D9008	SD1008	SD0	自诊断错误	自诊断错误号码	<ul style="list-style-type: none"> 当自诊断结果中发现错误时, 错误号码以 BIN 码存储 	○
D9009	SD1009	SD62	报警器检测	发生外部故障的 F 号码	<ul style="list-style-type: none"> 当 F0 到 255 中的一个被 OUTF 或 SETF 切换为 ON 时, 在已经变为 ON 的 F 号码中最先被检测到的 F 号码被以 BIN 码存储。 SD62 可以被 RSTF 或 LEDR 指令清除。如果另外一个 F 号码被检测到, SD62 的清除会引起下一个号码被存储进 SD62。 	QCPU Q2AS Q2A
					<ul style="list-style-type: none"> 当 F0 到 255 中的一个被 OUTF 或 SETF 切换为 ON 时, 在已经变为 ON 的 F 号码中最先被检测到的 F 号码被以 BIN 码存储。 可以通过执行 RSTF 或 LEDR 指令或者将 CPU 模块前面的 INDICATOR RESET 开关搬到 ON 位置, 来清除 SD62。 如果另外一个 F 号码被检测到, SD62 的清除会引起下一个号码被存储进 SD62。 	Q3A Q4A Q4AR
D9010	SD1010		错误步	发生操作错误步的 号码	<ul style="list-style-type: none"> 当在执行应用指令的过程中发生操作错误时, 发生错误的步的号码被以 BIN 码存储。此后, 每次发生操作错误时, SD1010 中的内容都被更新。 	○

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后特 殊寄存器	用于修改 特殊寄存器	名称	含义	详细内容	相应的 CPU
D9011	SD1011		错误步	发生错作错误的步 号码	● 当在执行应用指令的过程中发生操作错误时， 发生错误的步号码以 BIN 码存储。当 SM1011 从 OFF 变为 ON 时步号码被存储到 SD1011， SD1011 中的数据并未被更新，直到 SM1011 被用户程序清除。	○
D9014	SD1014		I/O 控制模式	I/O 控制模式号码	● I/O 控制模式设置被切换到以下任何一个号码： 0. 输入和输出到处于直接模式 1. 输入处于刷新模式，输出处于直接模式 3. 输入和输出都处于刷新模式	
D9015	SD1015	SD203	CPU 的工作状态	CPU 的工作状态	● 如下所示的 CPU 工作状态被存储在 SD203 中。 <div><div>b15 到 b12 b11 到 b8 b7 到 b4 b3 到 b0</div><div><div>通过计算机的远程 RUN/STOP</div><div><div>0 RUN</div><div>1 STOP</div><div>2 PAUSE *1</div></div></div><div><div>CPU 钥匙开关</div><div><div>0 RUN</div><div>1 STOP</div><div>2 PAUSE *1</div><div>3 STEP RUN</div></div><div>[保留和远程 RUN/STOP 模式的一样]</div><div><div>在程序中的状态</div><div><div>0 除下面以外的</div><div>1 <div>STOP</div>指令执行</div></div><div><div>通过参数设置的远程 RUN/STOP</div><div><div>0 RUN</div><div>1 STOP</div><div>2 PAUSE *1</div></div></div></div><p>* 1 当 CPU 模块处于 RUN 模式并且 SM1040 为 OFF 时，CPU 模块如果被改变到 PAUSE 模式也会保留在 RUN 模式。</p></div></div>	
D9016	SD1016		程序号码	0: 主程序 (ROM) 1: 主程序 (RAM) 2: 子程序 1 (RAM) 3: 子程序 2 (RAM) 4: 子程序 3 (RAM) 5: 子程序 1 (ROM) 6: 子程序 2 (ROM) 7: 子程序 3 (ROM) 8: 主程序 (E ² PROM) 9: 子程序 1 (E ² PROM) A: 子程序 2 (E ² PROM) B: 子程序 3 (E ² PROM)	● 表示当前正在运行的顺序程序，0 到 B 之间的一个 值被以 BIN 码存储。	
D9017	SD1017	SD520	扫描时间	最小扫描时间 (10ms 单位)	● 如果扫描时间小于 SD520 中的值，此值在每个 END 处被存储。也就是，扫描时间的最小值被以 BIN 码 存储到 SD520。	
D9018	SD1018	SD524	扫描时间	扫描时间 (10ms 单位)	● 扫描时间在每个 END 处被以 BIN 码存储，并保持经 常重写。	

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后特 殊寄存器	用于修改 特殊寄存器	名称	含义	详细内容	相应的 CPU
D9019	SD1019	SD526	扫描时间	最大扫描时间 (10ms 单位)	● 如果扫描时间大于 SD526 中的值, 此值在每个 END 处被存储。 也就是, 扫描时间的最大值被以 BIN 码存储到 SD526。	○
D9020	SD1020	<div></div>	固定扫描	固定扫描时间 (用户以 10ms 为单 位定义)	● 以 10ms 的倍数设置连续程序之间的间隔。 0: 无设置 1 到 200: 设置程序在 (设定值) × 10ms 的间隔上被执行。	
D9021	SD1021	—	扫描时间	扫描时间 (1ms 单位)	● 扫描时间以 BIN 码存储并在每个 END 后被更新。	
D9022	SD1022	SD412	1 秒计数器	以 1 秒为单位计数	● 当 PC CPU 开始启动时, 它开始每秒计数 1。 ● 计数器从 0 到 32767 开始计数, 然后跳到 -32768 再加到 0。计数器重复此计数周期。	
D9025	SD1025	SD210	时钟数据	时钟数据 (年, 月)	● 以 BCD 码存储年 (2 位低位数) 和月。 <div><div>b15 到 b12 b11 到 b8 b7 到 b4 b3 到 b0</div><div>1987, 七月 H8707</div><div>年 月</div></div>	
D9026	SD1026	SD211	时钟数据	时钟数据 (日期, 小时)	● 以 BCD 码存储日期和小时。 <div><div>b15 到 b12 b11 到 b8 b7 到 b4 b3 到 b0</div><div>31号, 10点钟 H3110</div><div>日期 小时</div></div>	
D9027	SD1027	SD212	时钟数据	时钟数据 (分钟, 秒)	● 以 BCD 码存储分钟和秒。 <div><div>b15 到 b12 b11 到 b8 b7 到 b4 b3 到 b0</div><div>35分钟, 48秒 H3548</div><div>分钟 秒</div></div>	
D9028	SD1028	SD213	时钟数据	时钟数据 (星期几)	● 以 BCD 码存储星期几。 <div><div>b15 到 b12 b11 到 b8 b7 到 b4 b3 到 b0</div><div>星期五 H0005</div><div>星期几</div><div>0 星期日 1 星期一 2 星期二 3 星期三 4 星期四 5 星期五 6 星期六</div><div>必须设置0</div></div>	
D9035	SD1035	SD648	扩展文件寄存器	使用块号码	● 以 BCD 码存储当前正在使用的扩展文件寄存器的块号码。	
D9036	SD1036	<div></div>	设备号码的扩展 文件寄存器指定	当来自扩展文件寄 存器的单个设备被 直接访问时设备号 码	● 在 SD1036 和 SD1037 上以 BIN 数据在 2 个字内, 指定用于扩展文件寄存器直接读写的设备号码。 从块号码 1 的 R0 开始到指定的设备号码, 使用连续号码。 <div><div>扩展文件寄存器</div><div>0 到 16383 块号码1区域 16384 到 块号码2区域</div><div>SD1037, SD1036 设备号码 (BIN 数据) 到</div></div>	
D9037	SD1037	<div></div>				
D9038	SD1038	SD207	LED 显示优先级	优先级 1 到 4	● 设置 ERROR LED 的优先级, 此 ERROR LED 点亮 (或闪烁) 以指示带错误代码号码的错误。 ● 优先级设置区域的组态如下所示。 <div><div>b15 到 b12 b11 到 b8 b7 到 b4 b3 到 b0</div><div>SD207 优先级4 优先级3 优先级2 优先级1 SD208 优先级7 优先级6 优先级5</div></div>	
D9039	SD1039	SD208		优先级 5 到 7	● 对于详细信息, 参见应用 CPU 用户手册和 ACPU 编程手册 (基础级)	

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后特 殊寄存器	用于修改 特殊寄存器	名称	含义	详细内容	相应的 CPU
D9044	SD1044		用于采样跟踪	采样过程中的步数 或时间	<ul style="list-style-type: none"> 外部设备切换为 ON/OFF。 在扫描 ----- 0 在时间 ----- 时间 (10 秒为单位) 以 BIN 码存储此值 	○
D9049	SD1049		用于 SFC 的工作 区	用于 SFC 的工作区	<ul style="list-style-type: none"> 以二进制值存储扩展文件寄存器块号码，此寄存器用作 SFC 程序执行的工作区域。 如果有 16K 字节或更小的空白区域被使用或者是 SM320 为 OFF 则存储“0”，但此空白区域不能为第一个扩展文件寄存器。 	
D9050	SD1050		SFC 程序错误号 码	SFC 程序产生的错 误代码	<ul style="list-style-type: none"> 以 BIN 码存储在 SFC 程序中发生的错误错误代码。 0: 无错误 80: SFC 程序参数错误 81: SFC 代码错误 82: 超出的同时执行的操作步数 83: 块启动错误 84: SFC 程序操作错误 	
D9051	SD1051		错误块	发生错误的块的号 码	<ul style="list-style-type: none"> 以 BIN 码存储在 SFC 程序中发生错误的块号码。 在错误 83 这种情况下，启动块的号码被存储。 	
D9052	SD1052			发生错误步的号码	<ul style="list-style-type: none"> 以 BIN 值存储在 SFC 程序中发生错误代码 84 的步的号码。 存储“0”，当错误代码 80, 81 或 82 发生时。 当错误代码 83 发生时，存储块启动步的号码。 	
D9053	SD1053		错误转换	错误发生处的转换 条件号码	<ul style="list-style-type: none"> 以 BIN 值存储在 SFC 程序中错误代码 84 发生处的转换条件号码。 存储“0”，当错误代码 80, 81, 82 或 83 发生时。 	
D9054	SD1054		错误顺序步	错误发生处的顺序 步号码	<ul style="list-style-type: none"> 以 BIN 码存储在 SFC 程序中发生错误 84 的转换条件的顺序步的号码和操作输出。 	
D9055	SD1055	SD812	状态闭锁	状态闭锁步	<ul style="list-style-type: none"> 存储状态闭锁发生时的步号码。 如果状态闭锁是发生在主顺序程序中，以二进制值存储步号码。 如果状态闭锁在 SFC 程序中被执行，存储块号码和步号码。 <div style="border: 1px solid black; padding: 5px; margin: 10px auto; width: fit-content;"> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px 5px;">块号码 (BIN)</div> <div style="border: 1px solid black; padding: 2px 5px;">块号码 (BIN)</div> </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> 高8位 低8位 </div> </div>	QnA
D9060	SD1060	SD392	软件版本	内部软件的软件版 本	<ul style="list-style-type: none"> 以 ASCII 码存储内部系统的软件版本。 <div style="display: flex; align-items: center; margin: 10px auto;"> <div style="border: 1px solid black; padding: 2px 10px;">高字节</div> <div style="border: 1px solid black; padding: 2px 10px;">低字节</div> </div> <p style="text-align: center; margin: 5px 0;">↑</p> <p>在低字节位置的数据是不确定的。 软件版本被存储在低字节位置。 对于版本“A”，举例来说，“41H”被存储。</p> <p>注意：初始系统的软件版本可能和打印在包装后面的版本信息指示的版本不一样。</p>	
D9072	SD1072		PLC 通讯检查	计算机链接数据检 查	<ul style="list-style-type: none"> 在串行通讯模块的自我回路返回测试中，串行通讯模块通过自动写 / 读数据进行通讯检查。 	
D9081	SD1081	SD714	在通讯请求登记 区域的空白块数 量	在通讯请求登记区 域的空白块数量	<ul style="list-style-type: none"> 将通讯请求登记区域中的空白块数量存储到和 MELSECNET/MINI-S3 主单元, A2CCPU 或 A52GCPU 相连的远程终端模块中。 	QnA

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后特 殊寄存器	用于修改 特殊寄存器	名称	含义	详细内容	相应的 CPU																																																																																																																
D9085	SD1085		用于设置时间检 查值的寄存器	默认值为 10s	<ul style="list-style-type: none">● 设置用于 MELSECNET/10 的数据链接指令 (ZNRD, ZNRW) 的检查时间。● 设置范围: 1s 到 65535s (1 到 65535)● 设置单位: 1s● 默认值: 10s (如果 0 被设定, 默认值 10s 就被应用。)																																																																																																																	
D9090	SD1090		额外的特殊功能 模块的数量	额外的特殊功能模 块的数量	<ul style="list-style-type: none">● 对于详细信息, 参考每个微型计算机程序包的手册。																																																																																																																	
D9091	SD1091		详细错误代码	自诊断详细错误代 码	<ul style="list-style-type: none">● 存储指令错误原因的详细代码。																																																																																																																	
D9094	SD1094	SD251	被替换的 I/O 模 块的头 I/O 号码	被替换的 I/O 模块 的头 I/O 号码	<ul style="list-style-type: none">● 以 BIN 值存储此 I/O 模块的头 I/O 号码的头两位数字, 此模块将被在线拆除/安装。 例子) 输入模块 X2F0 → H2F																																																																																																																	
D9100	SD1100	—	保险丝烧坏的模 块	以 16 个点为单元的 位模式, 表示了保 险丝被烧坏的模块	<ul style="list-style-type: none">● 保险丝被烧坏的输出模块号码 (以 16 数据点为单元), 按照位模式输入。(当参数设置被执行时, 预先设置输出单元号码) <div><table><tr><td>b15</td><td>b14</td><td>b13</td><td>b12</td><td>b11</td><td>b10</td><td>b9</td><td>b8</td><td>b7</td><td>b6</td><td>b5</td><td>b4</td><td>b3</td><td>b2</td><td>b1</td><td>b0</td></tr><tr><td>SD1100</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td></td><td></td><td></td><td></td><td>(YCD)</td><td></td><td></td><td></td><td>(YBD)</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>SD1101</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>SD1107</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table><p>↑ (Y_{CD}) (Y_{BD})</p><p>↑ 表示保险丝烧坏</p></div>	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1100	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0					(YCD)				(YBD)								SD1101	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	SD1107	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	
b15	b14				b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																																																				
SD1100	0				0	0	1	0	0	0	1	0	0	0	0	0	0	0																																																																																																				
							(YCD)				(YBD)																																																																																																											
SD1101	0				0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																																																				
SD1107	0				0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																																																				
D9101	SD1101				<ul style="list-style-type: none">● 对于远程 I/O 站点的输出模块也执行保险丝烧坏检查。 (如果正常状态恢复, 并不执行清除操作。因此, 需要由用户程序来执行清除操作。)																																																																																																																	
D9102	SD1102																																																																																																																					
D9103	SD1103																																																																																																																					
D9104	SD1104																																																																																																																					
D9105	SD1105																																																																																																																					
D9106	SD1106																																																																																																																					
D9107	SD1107																																																																																																																					
D9108	SD1108	—	步转换监视定时 器设置	定时器设置值和时 间到达时的 f 号码	<ul style="list-style-type: none">● 设置步转换看门狗定时器的设定值, 以及在看门狗定时器时间到达时变为 ON 的报警器号码 (F 号码) <div><table><tr><td>b15</td><td>到</td><td>b8</td><td>b7</td><td>到</td><td>b0</td></tr><tr><td colspan="6"><div><div></div><div></div></div></td></tr><tr><td colspan="3"></td><td>↑</td><td colspan="2">↑</td></tr><tr><td colspan="3"></td><td>定时器设置</td><td colspan="2">F 号码设置</td></tr><tr><td colspan="6">(1 到 255s, 以秒为单位)</td></tr></table><p>(通过使 MSM708 到 SM1114 中的任何一个变为 ON, 监视定时器启动。如果和定时器相对应的步之后的转换条件在设定时间内没有建立, 设定的报警器 (F) 变为 ON)</p></div>	b15	到	b8	b7	到	b0	<div><div></div><div></div></div>									↑	↑					定时器设置	F 号码设置		(1 到 255s, 以秒为单位)																																																																																								
b15	到				b8	b7	到	b0																																																																																																														
<div><div></div><div></div></div>																																																																																																																						
					↑	↑																																																																																																																
					定时器设置	F 号码设置																																																																																																																
(1 到 255s, 以秒为单位)																																																																																																																						
D9109	SD1109																																																																																																																					
D9110	SD1110																																																																																																																					
D9111	SD1111																																																																																																																					
D9112	SD1112																																																																																																																					
D9113	SD1113																																																																																																																					
D9114	SD1114																																																																																																																					
D9116	SD1116	—	I/O 模块验证错 误	以 16 数据点为单元 的位模式, 表示了有 验证错误的模块。	<ul style="list-style-type: none">● 当有 I/O 模块, 其数据和系统上电时输入的数据不一样, 被检测到时, 此 I/O 单元的号码 (以 16 数据点为单元) 被以位模式输入。(当执行参数设置时预先设置 I/O 单元号码) <div><table><tr><td>b15</td><td>b14</td><td>b13</td><td>b12</td><td>b11</td><td>b10</td><td>b9</td><td>b8</td><td>b7</td><td>b6</td><td>b5</td><td>b4</td><td>b3</td><td>b2</td><td>b1</td><td>b0</td></tr><tr><td>SD1116</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>(Y₀)</td></tr><tr><td>SD1117</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>SD1123</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table><p>↑ (Y₀) (Y₀)</p><p>↑ 表示 I/O 模块校验错误</p></div>	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1116	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1																(Y ₀)	SD1117	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	SD1123	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	
b15	b14				b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																																																				
SD1116	0				0	0	0	0	0	0	0	0	0	0	0	0	0	1																																																																																																				
																		(Y ₀)																																																																																																				
SD1117	0				0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																																																				
SD1123	0				0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																																																				
D9117	SD1117				<ul style="list-style-type: none">● 对远程 I/O 站点的模块也执行 I/O 模块验证检查。 (如果正常状态恢复, 并不执行清除操作。因此, 需要由用户程序来执行清除操作)																																																																																																																	
D9118	SD1118																																																																																																																					
D9119	SD1119																																																																																																																					
D9120	SD1120																																																																																																																					
D9121	SD1121																																																																																																																					
D9122	SD1122																																																																																																																					
D9123	SD1123																																																																																																																					

特殊寄存器列表(续表)

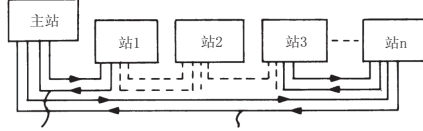
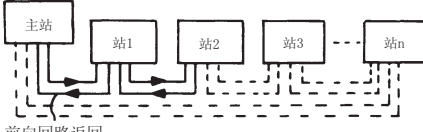
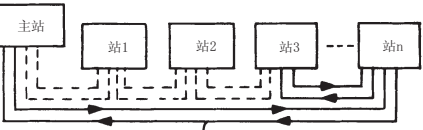
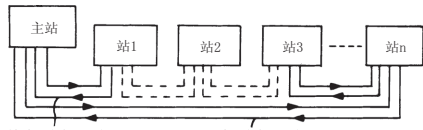
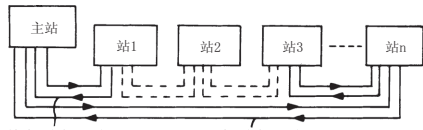
ACPU 特殊转换	转换后特 殊寄存器	用于修改 特殊寄存器	名称	含义	详细内容	相应的 CPU																																																																																																																																
D9124	SD1124	SD63	报警器探测数量	报警器探测数量	<ul style="list-style-type: none">当 F0 到 255 (对于 AuA 和 AnU 是 F0 到 247) 中的一个被 SETF 变为 ON 时, 1 被加到 SD63 的内容中。当 RSTF 或 LEDR 指令被执行时, 1 被从 SD63 的内容中减掉。(如果 CPU 模块提供了 INDICATOR RESET 开关的话, 按下这个开关可以执行同样的操作)被 SETF 变为 ON 的报警器的数量以 BIN 码存储到 SD63。SD63 的值最大是 8。																																																																																																																																	
D9125	SD1125	SD64	报警器探测号码	报警器探测号码	<ul style="list-style-type: none">当 F0 到 247 中的任何一个被 SETF 变为 ON 时, 报警器号码(F 号码)按照变为 ON 的顺序被登记到 D9125 到 D9132 中。被 RSTF 变为 OFF 的 F 号码被从 D9125 到 D9132 中擦除, 在被擦除的 F 号码移位到前面的寄存器后此 F 号码被存储。 <p>通过执行 LEDR 指令, SD64 到 SD71 中的内容向上移动一位。(对于 A3N, A3HCPU, 可以通过使用 CPU 模块前面的 INDICATOR RESET 开关来执行此操作)</p> <p>当有 8 个报警检测时, 如果第 9 个报警器被检测到, 它不会被存储到 SD64 到 SD71 中。</p>																																																																																																																																	
D9126	SD1126	SD65																																																																																																																																				
D9127	SD1127	SD66																																																																																																																																				
D9128	SD1128	SD67																																																																																																																																				
D9129	SD1129	SD68																																																																																																																																				
D9130	SD1130	SD69																																																																																																																																				
D9131	SD1131	SD70																																																																																																																																				
D9132	SD1132	SD71			<div><div>置位 置位 置位 RST 置位 置位 置位 置位 置位 置位 置位 置位</div><div>F50 F25 F99 F25 F15 F70 F65 F38 F110 F151 F210 LEDR</div><div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div><table><tr><td>SD62</td><td>0</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>99</td></tr><tr><td>SD63</td><td>0</td><td>1</td><td>2</td><td>3</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>8</td></tr><tr><td>SD64</td><td>0</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>99</td></tr><tr><td>SD65</td><td>0</td><td>0</td><td>25</td><td>25</td><td>99</td><td>99</td><td>99</td><td>99</td><td>99</td><td>99</td><td>99</td><td>15</td></tr><tr><td>SD66</td><td>0</td><td>0</td><td>0</td><td>99</td><td>0</td><td>15</td><td>15</td><td>15</td><td>15</td><td>15</td><td>15</td><td>70</td></tr><tr><td>SD67</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>70</td><td>70</td><td>70</td><td>70</td><td>70</td><td>65</td></tr><tr><td>SD68</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>65</td><td>65</td><td>65</td><td>65</td><td>38</td></tr><tr><td>SD69</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>38</td><td>38</td><td>38</td><td>110</td></tr><tr><td>SD70</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>110</td><td>110</td><td>151</td></tr><tr><td>SD71</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>151</td><td>210</td></tr></table></div>	SD62	0	50	50	50	50	50	50	50	50	50	50	99	SD63	0	1	2	3	2	3	4	5	6	7	8	8	SD64	0	50	50	50	50	50	50	50	50	50	50	99	SD65	0	0	25	25	99	99	99	99	99	99	99	15	SD66	0	0	0	99	0	15	15	15	15	15	15	70	SD67	0	0	0	0	0	0	70	70	70	70	70	65	SD68	0	0	0	0	0	0	0	65	65	65	65	38	SD69	0	0	0	0	0	0	0	0	38	38	38	110	SD70	0	0	0	0	0	0	0	0	0	110	110	151	SD71	0	0	0	0	0	0	0	0	0	0	151
SD62	0	50	50	50	50	50	50	50	50	50	50	99																																																																																																																										
SD63	0	1	2	3	2	3	4	5	6	7	8	8																																																																																																																										
SD64	0	50	50	50	50	50	50	50	50	50	50	99																																																																																																																										
SD65	0	0	25	25	99	99	99	99	99	99	99	15																																																																																																																										
SD66	0	0	0	99	0	15	15	15	15	15	15	70																																																																																																																										
SD67	0	0	0	0	0	0	70	70	70	70	70	65																																																																																																																										
SD68	0	0	0	0	0	0	0	65	65	65	65	38																																																																																																																										
SD69	0	0	0	0	0	0	0	0	38	38	38	110																																																																																																																										
SD70	0	0	0	0	0	0	0	0	0	110	110	151																																																																																																																										
SD71	0	0	0	0	0	0	0	0	0	0	151	210																																																																																																																										

特殊寄存器列表

(10) 特殊寄存器列表，专用于 QnA

ACPU 特殊转换	转换后特 殊寄存器	用于修改 特殊寄存器	名称	含义	详细内容	相应的 CPU																																																																																				
D9200	SD1200	—	ZNRD (用于 ACPU 的 LRDP)处理结果	0:正常结束 2:ZNRD 指令设置故障 3:相关站上的错误 4:相关站的 ZNRD 执行被禁止	存储 ZNRD (字设备读)指令的执行结果。 ● ZNRD 指令设置故障: ZNRD 指令的常量, 源端点, 和 / 或目标端点的故障设置。 ● 相应站错误: 其中的一个站无法通信。 ● ZNRD 不能在相应站上执行: 指定的站是一个远程 I/O 站	QnA																																																																																				
D9201	SD1201	—	ZNWR (用于 ACPU 的 LWTP)处理结果	0:正常结束 2:ZNWR 指令设置故障 3:相关站上的错误 4:相关站的 ZNWR 执行被禁止	存储 ZNW R (字设备写)指令的执行结果。 ● ZNW R 指令设置故障: ZNW R 指令的常量, 源端点, 和 / 或目标端点的故障设置。 ● 相应站错误: 其中的一个站无法通信。 ● ZNW R 不能在相应站上执行: 指定的站是一个远程 I/O 站																																																																																					
D9202	SD1202	—	本地站链接类型	存储号码 1 到 16 的条件	存储从站是否对应于 MELSECNET 或 MELSECNET11。 ● 和 MELSECNET 11 站对应的位变为“1。” ● 和 MELSECNET 站对应的位或者没有连接上的位变为“0。” <table><tr><th>设备号</th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr><tr><td>SD1202</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr><tr><td>SD1203</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr><tr><td>SD1241</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr><tr><td>SD1242</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr></table>		设备号	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1202	L16	L15	L14	L13	L12	L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1	SD1203	L32	L31	L30	L29	L28	L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17	SD1241	L48	L47	L46	L45	L44	L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33	SD1242	L64	L63	L62	L61	L60	L59	L58	L57	L56	L55	L54	L53	L52	L51	L50
设备号	b15	b14		b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																									
SD1202	L16	L15	L14	L13	L12	L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1																																																																										
SD1203	L32	L31	L30	L29	L28	L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17																																																																										
SD1241	L48	L47	L46	L45	L44	L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33																																																																										
SD1242	L64	L63	L62	L61	L60	L59	L58	L57	L56	L55	L54	L53	L52	L51	L50	L49																																																																										
D9203	SD1203	—		存储号码 17 到 32 的条件	● 如果在操作过程中一个本地站停止, 停止之前的内容被保持。 SD1224 到 SD1227 和 SD1228 到 SD1231 中的内容都被执行 OR 处理。如果相应的位为“0”, 上面的特殊寄存器中的相应位变为有效。 ● 如果自站(主站)停止, 停止之前的内容也被保持。																																																																																					
D9204	SD1204	—	链接状态	0:前向回路, 处于数据链接中 1:反向回路, 处于数据链接中 2:前向 / 反向方向中的回路返回实现 3:只在前向方向实现回路返回 4:只在反向方向实现回路返回 5:数据链接禁止	存储数据链接的当前路径状态 ● 前向回路中的数据链接 ● 反向回路中的数据链接 	QnA																																																																																				

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后特 殊寄存器	用于修改 特殊寄存器	名称	含义	详细内容	相应的 CPU																																																																																																						
D9204	SD1204	—	链接状态	0:前向回路,处于数 据链接中 1:反向回路,处于数 据链接中 2:前向 / 反向方向中 的回路返回实现 3:只在前向方向实现 回路返回 4:只在反向方向实现 回路返回 5:数据链接禁止	<div><div><div>● 前向 / 反向回路中的回路返回</div><div><div>前向回路返回 反向回路返回</div></div><div>● 只在前向回路中回路返回</div><div><div>前向回路返回</div></div><div>● 只在反向回路中回路返回</div><div><div>反向回路返回</div></div></div></div> <div><div>存储回路返回被执行的本地或者远程 I/O 站的号码。</div><div><div>前向回路返回 反向回路返回</div></div></div>	QnA																																																																																																						
D9205	SD1205	—	站实现回路返回	实现前向回路返回的 站	<div><div>存储回路返回被执行的本地或者远程 I/O 站的号码。</div><div><div>前向回路返回 反向回路返回</div></div></div>																																																																																																							
D9206	SD1206	—	站实现回路返回	实现反向回路返回的 站	<div><div>在上面的例子中, 1 被存储到 D9205, 并且 3 被存储到 D9206。如果数据链接恢复到正常状态 (前向回路中的数据链接), D9205 和 D9206 中的值保持为 1 和 3。因此, 要想将它们恢复到“0”, 使用顺序程序或者执行复位操作。</div></div>																																																																																																							
D9210	SD1210	—	重试次数	作为累加值存储	存储由于传输错误引起的重试次数。 计数在最大值“FFFFh”处停止。 要想恢复到“0”值, 执行复位操作。																																																																																																							
D9211	SD1211	—	回路被选择的次 数	作为累加值存储	存储回路线路被切换到反向回路或回路返回的次数。 计数在最大值“FFFFh”处停止。 要想恢复到“0”值, 执行复位操作。																																																																																																							
D9212	SD1212	—	本地站操作状态	存储号码 1 到 16 的 条件	存储处于 STOP 或 PAUSE 模式的本地站的号码	QnA																																																																																																						
D9213	SD1213	—	本地站操作状态	存储号码 17 到 32 的 条件	<table><tr><th>设备号</th><th colspan="16">位</th></tr><tr><th></th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr><tr><td>SD1212</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr><tr><td>SD1213</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr><tr><td>SD1214</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr><tr><td>SD1215</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr></table>		设备号	位																	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1212	L16	L15	L14	L13	L12	L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1	SD1213	L32	L31	L30	L29	L28	L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17	SD1214	L48	L47	L46	L45	L44	L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33	SD1215	L64	L63	L62	L61	L60	L59	L58	L57	L56	L55	L54	L53	L52	L51	L50	L49
设备号	位																																																																																																											
	b15	b14	b13	b12	b11		b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																																											
SD1212	L16	L15	L14	L13	L12	L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1																																																																																												
SD1213	L32	L31	L30	L29	L28	L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17																																																																																												
SD1214	L48	L47	L46	L45	L44	L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33																																																																																												
SD1215	L64	L63	L62	L61	L60	L59	L58	L57	L56	L55	L54	L53	L52	L51	L50	L49																																																																																												
D9214	SD1214	—	本地站操作状态	存储号码 33 到 48 的 条件																																																																																																								
D9215	SD1215	—	本地站操作状态	存储号码 49 到 64 的 条件	当一个本地站被切换到 STOP 或 PAUSE 模式, 在寄存器中和此站号码相对应的位变为“1” 例子: 当站 7 切换到 STOP 模式, SD1212 中的 b6 变为“1”, 并且当 SD1212 被监视时, 它的值为“64 (40h)”																																																																																																							

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后特 殊寄存器	用于修改 特殊寄存器	名称	含义	详细内容	相应的 CPU																																																																																																					
D9216	SD1216	—	本站站错误探测 状态	存储号码 1 到 16 的条件	存储处于错误中的本站站号码 <table><tr><th rowspan="2">设备号</th><th colspan="16">位</th></tr><tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr><tr><td>SD1216</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr><tr><td>SD1217</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr><tr><td>SD1218</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr><tr><td>SD1219</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr></table>	设备号	位																b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1216	L16	L15	L14	L13	L12	L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1	SD1217	L32	L31	L30	L29	L28	L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17	SD1218	L48	L47	L46	L45	L44	L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33	SD1219	L64	L63	L62	L61	L60	L59	L58	L57	L56	L55	L54	L53	L52	L51	L50	L49	QnA
设备号	位																																																																																																										
	b15	b14		b13		b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																																									
SD1216	L16	L15		L14		L13	L12	L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1																																																																																									
SD1217	L32	L31	L30	L29	L28	L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17																																																																																											
SD1218	L48	L47	L46	L45	L44	L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33																																																																																											
SD1219	L64	L63	L62	L61	L60	L59	L58	L57	L56	L55	L54	L53	L52	L51	L50	L49																																																																																											
D9217	SD1217	—	存储号码 17 到 32 的条件																																																																																																								
D9218	SD1218	—	存储号码 33 到 48 的条件																																																																																																								
D9219	SD1219	—	存储号码 49 到 64 的条件	如果本站站探测到错误，和站号码相对应的位变为“1”。 例子：当站 6 和 12 探测到错误，SD1216 中的 b5 和 11 变为“1”，并且当 SD1216 被监视时，它的值为“2080 (820h)”																																																																																																							
D9220	SD1220	—	本站站参数未确 认；远程 I/O 站 I/O 分配错误	存储号码 1 到 16 的条件	存储包含不匹配参数的本站站号码，或者是有不正确的 I/O 分配的远程站号码。 <table><tr><th rowspan="2">设备号</th><th colspan="16">位</th></tr><tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr><tr><td>SD1220</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr><tr><td>SD1221</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr><tr><td>SD1222</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr><tr><td>SD1223</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr></table>	设备号	位																b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1220	L16	L15	L14	L13	L12	L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1	SD1221	L32	L31	L30	L29	L28	L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17	SD1222	L48	L47	L46	L45	L44	L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33	SD1223	L64	L63	L62	L61	L60	L59	L58	L57	L56	L55	L54	L53	L52	L51	L50	L49	QnA
设备号	位																																																																																																										
	b15	b14		b13		b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																																									
SD1220	L16	L15		L14		L13	L12	L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1																																																																																									
SD1221	L32	L31	L30	L29	L28	L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17																																																																																											
SD1222	L48	L47	L46	L45	L44	L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33																																																																																											
SD1223	L64	L63	L62	L61	L60	L59	L58	L57	L56	L55	L54	L53	L52	L51	L50	L49																																																																																											
D9221	SD1221	—	存储号码 17 到 32 的条件																																																																																																								
D9222	SD1222	—	存储号码 33 到 48 的条件																																																																																																								
D9223	SD1223	—	存储号码 49 到 64 的条件	如果一个在一个三级网络中作为主站运行的本站站探测到参数错误，或者一个远程 I/O 站的 I/O 分配不正常，和此本站站或者远程 I/O 站相对应的设备号码的位变为“1”。 例子：当本站站 5 和远程 I/O 站 14 探测到错误时，SD1220 中的 b4 和 b13 变为“1”，并且当 SD1220 被监视时，它的值为“8208 (2010h)”																																																																																																							
D9224	SD1224	—	本站站和远程 I/O 站初始化通 信进行中	存储号码 1 到 16 的条件	存储本地或者远程站的号码，当这些站正在和它们的相关主站进行初始化数据通信时。 <table><tr><th rowspan="2">设备号</th><th colspan="16">位</th></tr><tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr><tr><td>SD1224</td><td>LR 16</td><td>LR 15</td><td>LR 14</td><td>LR 13</td><td>LR 12</td><td>LR 11</td><td>LR 10</td><td>LR 9</td><td>LR 8</td><td>LR 7</td><td>LR 6</td><td>LR 5</td><td>LR 4</td><td>LR 3</td><td>LR 2</td><td>LR 1</td></tr><tr><td>SD1225</td><td>LR 32</td><td>LR 31</td><td>LR 30</td><td>LR 29</td><td>LR 28</td><td>LR 27</td><td>LR 26</td><td>LR 25</td><td>LR 24</td><td>LR 23</td><td>LR 22</td><td>LR 21</td><td>LR 20</td><td>LR 19</td><td>LR 18</td><td>LR 17</td></tr><tr><td>SD1226</td><td>LR 48</td><td>LR 47</td><td>LR 46</td><td>LR 45</td><td>LR 44</td><td>LR 43</td><td>LR 42</td><td>LR 41</td><td>LR 40</td><td>LR 39</td><td>LR 38</td><td>LR 37</td><td>LR 36</td><td>LR 35</td><td>LR 34</td><td>LR 33</td></tr><tr><td>SD1227</td><td>LR 64</td><td>LR 63</td><td>LR 62</td><td>LR 61</td><td>LR 60</td><td>LR 59</td><td>LR 58</td><td>LR 57</td><td>LR 56</td><td>LR 55</td><td>LR 54</td><td>LR 53</td><td>LR 52</td><td>LR 51</td><td>LR 50</td><td>LR 49</td></tr></table>	设备号	位																b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1224	LR 16	LR 15	LR 14	LR 13	LR 12	LR 11	LR 10	LR 9	LR 8	LR 7	LR 6	LR 5	LR 4	LR 3	LR 2	LR 1	SD1225	LR 32	LR 31	LR 30	LR 29	LR 28	LR 27	LR 26	LR 25	LR 24	LR 23	LR 22	LR 21	LR 20	LR 19	LR 18	LR 17	SD1226	LR 48	LR 47	LR 46	LR 45	LR 44	LR 43	LR 42	LR 41	LR 40	LR 39	LR 38	LR 37	LR 36	LR 35	LR 34	LR 33	SD1227	LR 64	LR 63	LR 62	LR 61	LR 60	LR 59	LR 58	LR 57	LR 56	LR 55	LR 54	LR 53	LR 52	LR 51	LR 50	LR 49	QnA
设备号	位																																																																																																										
	b15	b14		b13		b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																																									
SD1224	LR 16	LR 15		LR 14		LR 13	LR 12	LR 11	LR 10	LR 9	LR 8	LR 7	LR 6	LR 5	LR 4	LR 3	LR 2	LR 1																																																																																									
SD1225	LR 32	LR 31	LR 30	LR 29	LR 28	LR 27	LR 26	LR 25	LR 24	LR 23	LR 22	LR 21	LR 20	LR 19	LR 18	LR 17																																																																																											
SD1226	LR 48	LR 47	LR 46	LR 45	LR 44	LR 43	LR 42	LR 41	LR 40	LR 39	LR 38	LR 37	LR 36	LR 35	LR 34	LR 33																																																																																											
SD1227	LR 64	LR 63	LR 62	LR 61	LR 60	LR 59	LR 58	LR 57	LR 56	LR 55	LR 54	LR 53	LR 52	LR 51	LR 50	LR 49																																																																																											
D9225	SD1225	—	存储号码 17 到 32 的条件																																																																																																								
D9226	SD1226	—	存储号码 33 到 48 的条件																																																																																																								
D9227	SD1227	—	存储号码 49 到 64 的条件	和正在通信初始化设置的站号码相对应的位变为“1”。 例子：当站 23 和 45 正在通信，SD1225 中的 b6 和 SD1226 中的 b12 变为“1”，并且当 SD1225 被监视时，它的值位“64 (40h)”，当 SD1226 被监视时，它的值为“4096 (1000h)”																																																																																																							
D9228	SD1228	—	本站站和远程 I/O 站错误	存储号码 1 到 16 的条件	存储处于错误中的本地或远程站号码。 <table><tr><th rowspan="2">设备号</th><th colspan="16">位</th></tr><tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr><tr><td>SD1228</td><td>LR 16</td><td>LR 15</td><td>LR 14</td><td>LR 13</td><td>LR 12</td><td>LR 11</td><td>LR 10</td><td>LR 9</td><td>LR 8</td><td>LR 7</td><td>LR 6</td><td>LR 5</td><td>LR 4</td><td>LR 3</td><td>LR 2</td><td>LR 1</td></tr><tr><td>SD1229</td><td>LR 32</td><td>LR 31</td><td>LR 30</td><td>LR 29</td><td>LR 28</td><td>LR 27</td><td>LR 26</td><td>LR 25</td><td>LR 24</td><td>LR 23</td><td>LR 22</td><td>LR 21</td><td>LR 20</td><td>LR 19</td><td>LR 18</td><td>LR 17</td></tr><tr><td>SD1230</td><td>LR 48</td><td>LR 47</td><td>LR 46</td><td>LR 45</td><td>LR 44</td><td>LR 43</td><td>LR 42</td><td>LR 41</td><td>LR 40</td><td>LR 39</td><td>LR 38</td><td>LR 37</td><td>LR 36</td><td>LR 35</td><td>LR 34</td><td>LR 33</td></tr><tr><td>SD1231</td><td>LR 64</td><td>LR 63</td><td>LR 62</td><td>LR 61</td><td>LR 60</td><td>LR 59</td><td>LR 58</td><td>LR 57</td><td>LR 56</td><td>LR 55</td><td>LR 54</td><td>LR 53</td><td>LR 52</td><td>LR 51</td><td>LR 50</td><td>LR 49</td></tr></table>	设备号	位																b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1228	LR 16	LR 15	LR 14	LR 13	LR 12	LR 11	LR 10	LR 9	LR 8	LR 7	LR 6	LR 5	LR 4	LR 3	LR 2	LR 1	SD1229	LR 32	LR 31	LR 30	LR 29	LR 28	LR 27	LR 26	LR 25	LR 24	LR 23	LR 22	LR 21	LR 20	LR 19	LR 18	LR 17	SD1230	LR 48	LR 47	LR 46	LR 45	LR 44	LR 43	LR 42	LR 41	LR 40	LR 39	LR 38	LR 37	LR 36	LR 35	LR 34	LR 33	SD1231	LR 64	LR 63	LR 62	LR 61	LR 60	LR 59	LR 58	LR 57	LR 56	LR 55	LR 54	LR 53	LR 52	LR 51	LR 50	LR 49	QnA
设备号	位																																																																																																										
	b15	b14		b13		b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																																									
SD1228	LR 16	LR 15		LR 14		LR 13	LR 12	LR 11	LR 10	LR 9	LR 8	LR 7	LR 6	LR 5	LR 4	LR 3	LR 2	LR 1																																																																																									
SD1229	LR 32	LR 31	LR 30	LR 29	LR 28	LR 27	LR 26	LR 25	LR 24	LR 23	LR 22	LR 21	LR 20	LR 19	LR 18	LR 17																																																																																											
SD1230	LR 48	LR 47	LR 46	LR 45	LR 44	LR 43	LR 42	LR 41	LR 40	LR 39	LR 38	LR 37	LR 36	LR 35	LR 34	LR 33																																																																																											
SD1231	LR 64	LR 63	LR 62	LR 61	LR 60	LR 59	LR 58	LR 57	LR 56	LR 55	LR 54	LR 53	LR 52	LR 51	LR 50	LR 49																																																																																											
D9229	SD1229	—	存储号码 17 到 32 的条件																																																																																																								
D9230	SD1230	—	存储号码 33 到 48 的条件																																																																																																								
D9231	SD1231	—	存储号码 49 到 64 的条件	和带有错误的站号码相对应的位变为“1”。 例子：当本站站 3 和远程 I/O 站 14 有错误时，SD1228 的 b2 和 b13 变为“1”，并且当 SD1228 被监视时，它的值为“8196 (2004h)”																																																																																																							

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后特 殊寄存器	用于修改 特殊寄存器	名称	含义	详细内容	相应的 CPU																																																																																																																																																																																																																																																																																																																																														
D9232	SD1232	—	本站和远程 I/O 站回路错误	存储号码 1 到 8 的条件	存储发生了前向或者反向回路错误的本地或远程站号码。 <table><tr><th>设备号</th><th colspan="16">位</th></tr><tr><th></th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr><tr><td>SD1232</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td></tr><tr><td></td><td>L/R8</td><td>L/R7</td><td>L/R6</td><td>L/R5</td><td>L/R4</td><td>L/R3</td><td>L/R2</td><td>L/R1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>SD1233</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td></tr><tr><td></td><td>L/R16</td><td>L/R15</td><td>L/R14</td><td>L/R13</td><td>L/R12</td><td>L/R11</td><td>L/R10</td><td>L/R9</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>SD1234</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td></tr><tr><td></td><td>L/R24</td><td>L/R23</td><td>L/R22</td><td>L/R21</td><td>L/R20</td><td>L/R19</td><td>L/R18</td><td>L/R17</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>SD1235</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td></tr><tr><td></td><td>L/R32</td><td>L/R31</td><td>L/R30</td><td>L/R29</td><td>L/R28</td><td>L/R27</td><td>L/R26</td><td>L/R25</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>D9236</td><td>SD1236</td><td>—</td><td></td><td>存储号码 33 到 40 的条件</td><td rowspan="10">在上面的表中，“F”表示前向回路线路，“R”表示一个反向回路线路。和发生前向或反向回路错误的站号码相对应的位变为“1”。 例子：当站 5 的前向回路线路发生错误时，SD1232 的 b8 变为“1”，并且当 SD1232 被监视时，它的值变为“256(100h)”</td><td rowspan="10">QnA</td></tr><tr><td>D9237</td><td>SD1237</td><td>—</td><td></td><td>存储号码 41 到 48 的条件</td></tr><tr><td>D9238</td><td>SD1238</td><td>—</td><td></td><td>存储号码 49 到 56 的条件</td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td>D9239</td><td>SD1239</td><td>—</td><td></td><td>存储号码 57 到 64 的条件</td></tr><tr><td>D9240</td><td>SD1240</td><td>—</td><td>通信错误检测到的次数</td><td>存储接收到错误的次数的累加值</td><td>存储下列传输错误被检测到的次数： CRC, OVER, AB, IF 计数最大到 FFFFh。要想恢复到“0”值，执行复位操作。</td></tr><tr><td>D9241</td><td>SD1241</td><td>—</td><td rowspan="3">本站站链接类型</td><td>存储号码 33 到 48 的条件</td><td>存储此从站是否对应于 MELSECNET 或 MELSECNET11。 ● 和 MELSECNET 11 站相对应的位变为“1。” ● 和 MELSECNET 站相对应的位或者没有被连接的位变为“0。” <table><tr><th>设备号</th><th colspan="16">位</th></tr><tr><th></th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr><tr><td>SD1202</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr><tr><td>SD1203</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr><tr><td>SD1241</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr><tr><td>SD1242</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr></table></td></tr><tr><td>D9242</td><td>SD1242</td><td>—</td><td></td><td>存储号码 49 到 64 的条件</td><td>● 如果一个本站站在操作过程中停止，停止前的内容被保持。 SD1224 到 SD1227 和 SD1228 到 SD1231 中的内容被执行 OR 处理。如果相应的位为“0”，和上面的特殊寄存器相对应的位变为有效。 ● 如果本站站（主站）停止，停止之前的内容也被保持。</td></tr><tr><td>D9243</td><td>SD1243</td><td>—</td><td>用于本站的站号码信息</td><td>存储站号码(0 到 64)</td><td>允许一个本站站去确认它自己的站号码。</td></tr><tr><td>D9244</td><td>SD1244</td><td>—</td><td>连接设备站的数量</td><td>存储从站的数量</td><td>表示在一个回路中的从站的数量。</td></tr><tr><td>D9245</td><td>SD1245</td><td>—</td><td>通信错误检测到的次数</td><td>存储接收到错误的次数的累加值</td><td>存储下列传输错误 S 被检测到的次数：CRC, OVER, AB, IF 计数最大到 FFFFh。要想恢复到“0”值，执行复位操作。</td></tr></table>	设备号	位																	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1232	R	F	R	F	R	F	R	F	R	F	R	F	R	F	R	F		L/R8	L/R7	L/R6	L/R5	L/R4	L/R3	L/R2	L/R1									SD1233	R	F	R	F	R	F	R	F	R	F	R	F	R	F	R	F		L/R16	L/R15	L/R14	L/R13	L/R12	L/R11	L/R10	L/R9									SD1234	R	F	R	F	R	F	R	F	R	F	R	F	R	F	R	F		L/R24	L/R23	L/R22	L/R21	L/R20	L/R19	L/R18	L/R17									SD1235	R	F	R	F	R	F	R	F	R	F	R	F	R	F	R	F		L/R32	L/R31	L/R30	L/R29	L/R28	L/R27	L/R26	L/R25									D9236	SD1236	—		存储号码 33 到 40 的条件	在上面的表中，“F”表示前向回路线路，“R”表示一个反向回路线路。和发生前向或反向回路错误的站号码相对应的位变为“1”。 例子：当站 5 的前向回路线路发生错误时，SD1232 的 b8 变为“1”，并且当 SD1232 被监视时，它的值变为“256(100h)”	QnA	D9237	SD1237	—		存储号码 41 到 48 的条件	D9238	SD1238	—		存储号码 49 到 56 的条件						D9239	SD1239	—		存储号码 57 到 64 的条件	D9240	SD1240	—	通信错误检测到的次数	存储接收到错误的次数的累加值	存储下列传输错误被检测到的次数： CRC, OVER, AB, IF 计数最大到 FFFFh。要想恢复到“0”值，执行复位操作。	D9241	SD1241	—	本站站链接类型	存储号码 33 到 48 的条件	存储此从站是否对应于 MELSECNET 或 MELSECNET11。 ● 和 MELSECNET 11 站相对应的位变为“1。” ● 和 MELSECNET 站相对应的位或者没有被连接的位变为“0。” <table><tr><th>设备号</th><th colspan="16">位</th></tr><tr><th></th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr><tr><td>SD1202</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr><tr><td>SD1203</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr><tr><td>SD1241</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr><tr><td>SD1242</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr></table>	设备号	位																	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1202	L16	L15	L14	L13	L12	L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1	SD1203	L32	L31	L30	L29	L28	L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17	SD1241	L48	L47	L46	L45	L44	L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33	SD1242	L64	L63	L62	L61	L60	L59	L58	L57	L56	L55	L54	L53	L52	L51	L50	L49	D9242	SD1242	—		存储号码 49 到 64 的条件	● 如果一个本站站在操作过程中停止，停止前的内容被保持。 SD1224 到 SD1227 和 SD1228 到 SD1231 中的内容被执行 OR 处理。如果相应的位为“0”，和上面的特殊寄存器相对应的位变为有效。 ● 如果本站站（主站）停止，停止之前的内容也被保持。	D9243	SD1243	—	用于本站的站号码信息	存储站号码(0 到 64)	允许一个本站站去确认它自己的站号码。	D9244	SD1244	—	连接设备站的数量	存储从站的数量	表示在一个回路中的从站的数量。	D9245	SD1245	—	通信错误检测到的次数	存储接收到错误的次数的累加值	存储下列传输错误 S 被检测到的次数：CRC, OVER, AB, IF 计数最大到 FFFFh。要想恢复到“0”值，执行复位操作。
设备号	位																																																																																																																																																																																																																																																																																																																																																			
	b15	b14		b13		b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																																																																																																																																																																																																																																																																																		
SD1232	R	F		R		F	R	F	R	F	R	F	R	F	R	F	R	F																																																																																																																																																																																																																																																																																																																																		
	L/R8	L/R7		L/R6		L/R5	L/R4	L/R3	L/R2	L/R1																																																																																																																																																																																																																																																																																																																																										
SD1233	R	F		R		F	R	F	R	F	R	F	R	F	R	F	R	F																																																																																																																																																																																																																																																																																																																																		
	L/R16	L/R15		L/R14		L/R13	L/R12	L/R11	L/R10	L/R9																																																																																																																																																																																																																																																																																																																																										
SD1234	R	F		R		F	R	F	R	F	R	F	R	F	R	F	R	F																																																																																																																																																																																																																																																																																																																																		
	L/R24	L/R23		L/R22		L/R21	L/R20	L/R19	L/R18	L/R17																																																																																																																																																																																																																																																																																																																																										
SD1235	R	F		R		F	R	F	R	F	R	F	R	F	R	F	R	F																																																																																																																																																																																																																																																																																																																																		
	L/R32	L/R31	L/R30	L/R29	L/R28	L/R27	L/R26	L/R25																																																																																																																																																																																																																																																																																																																																												
D9236	SD1236	—		存储号码 33 到 40 的条件	在上面的表中，“F”表示前向回路线路，“R”表示一个反向回路线路。和发生前向或反向回路错误的站号码相对应的位变为“1”。 例子：当站 5 的前向回路线路发生错误时，SD1232 的 b8 变为“1”，并且当 SD1232 被监视时，它的值变为“256(100h)”	QnA																																																																																																																																																																																																																																																																																																																																														
D9237	SD1237	—		存储号码 41 到 48 的条件																																																																																																																																																																																																																																																																																																																																																
D9238	SD1238	—		存储号码 49 到 56 的条件																																																																																																																																																																																																																																																																																																																																																
D9239	SD1239	—		存储号码 57 到 64 的条件																																																																																																																																																																																																																																																																																																																																																
D9240	SD1240	—	通信错误检测到的次数	存储接收到错误的次数的累加值			存储下列传输错误被检测到的次数： CRC, OVER, AB, IF 计数最大到 FFFFh。要想恢复到“0”值，执行复位操作。																																																																																																																																																																																																																																																																																																																																													
D9241	SD1241	—	本站站链接类型	存储号码 33 到 48 的条件			存储此从站是否对应于 MELSECNET 或 MELSECNET11。 ● 和 MELSECNET 11 站相对应的位变为“1。” ● 和 MELSECNET 站相对应的位或者没有被连接的位变为“0。” <table><tr><th>设备号</th><th colspan="16">位</th></tr><tr><th></th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr><tr><td>SD1202</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr><tr><td>SD1203</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr><tr><td>SD1241</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr><tr><td>SD1242</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr></table>	设备号	位																	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1202	L16	L15	L14	L13	L12	L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1	SD1203	L32	L31	L30	L29	L28	L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17	SD1241	L48	L47	L46	L45	L44	L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33	SD1242	L64	L63	L62	L61	L60	L59	L58	L57	L56	L55	L54	L53	L52	L51	L50	L49																																																																																																																																																																																																																																							
设备号	位																																																																																																																																																																																																																																																																																																																																																			
	b15	b14		b13			b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																																																																																																																																																																																																																																																																																	
SD1202	L16	L15	L14	L13			L12	L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1																																																																																																																																																																																																																																																																																																																																		
SD1203	L32	L31	L30	L29	L28	L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17																																																																																																																																																																																																																																																																																																																																				
SD1241	L48	L47	L46	L45	L44	L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33																																																																																																																																																																																																																																																																																																																																				
SD1242	L64	L63	L62	L61	L60	L59	L58	L57	L56	L55	L54	L53	L52	L51	L50	L49																																																																																																																																																																																																																																																																																																																																				
D9242	SD1242	—		存储号码 49 到 64 的条件	● 如果一个本站站在操作过程中停止，停止前的内容被保持。 SD1224 到 SD1227 和 SD1228 到 SD1231 中的内容被执行 OR 处理。如果相应的位为“0”，和上面的特殊寄存器相对应的位变为有效。 ● 如果本站站（主站）停止，停止之前的内容也被保持。																																																																																																																																																																																																																																																																																																																																															
D9243	SD1243	—	用于本站的站号码信息	存储站号码(0 到 64)	允许一个本站站去确认它自己的站号码。																																																																																																																																																																																																																																																																																																																																															
D9244	SD1244	—	连接设备站的数量	存储从站的数量	表示在一个回路中的从站的数量。																																																																																																																																																																																																																																																																																																																																															
D9245	SD1245	—	通信错误检测到的次数	存储接收到错误的次数的累加值	存储下列传输错误 S 被检测到的次数：CRC, OVER, AB, IF 计数最大到 FFFFh。要想恢复到“0”值，执行复位操作。																																																																																																																																																																																																																																																																																																																																															

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后特 殊寄存器	用于修改 特殊寄存器	名称	含义	详细内容	相应的 CPU																																																																																																					
D9248	SD1248	—	本站站操作状 态	存储号码 1 到 16 的 条件	存储处于 STOP 或 PAUSE 模式中的本站站号码	QnA																																																																																																					
D9249	SD1249	—		存储号码 17 到 32 的条件	<table><tr><th rowspan="2">设备号</th><th colspan="16">位</th></tr><tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr><tr><td>SD1248</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr><tr><td>SD1249</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr><tr><td>SD1250</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr><tr><td>SD1251</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr></table>		设备号	位																b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1248	L16	L15	L14	L13	L12	L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1	SD1249	L32	L31	L30	L29	L28	L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17	SD1250	L48	L47	L46	L45	L44	L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33	SD1251	L64	L63	L62	L61	L60	L59	L58	L57	L56	L55	L54	L53	L52	L51	L50	L49
设备号	位																																																																																																										
	b15	b14		b13	b12		b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																																									
SD1248	L16	L15	L14	L13	L12		L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1																																																																																										
SD1249	L32	L31	L30	L29	L28		L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17																																																																																										
SD1250	L48	L47	L46	L45	L44		L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33																																																																																										
SD1251	L64	L63	L62	L61	L60		L59	L58	L57	L56	L55	L54	L53	L52	L51	L50	L49																																																																																										
D9250	SD1250	—	存储号码 33 到 48 的条件																																																																																																								
D9251	SD1251	—	存储号码 49 到 64 的条件	和处于 STOP 或 PAUSE 模式的站号码相对应的位变为“1”。 例子：当本站站 7 和 15 处于 STOP 模式，SD1248 的 b6 和 b14 变为“1”，并且当 SD1248 被监视时，它的值为“16448(4040h)”																																																																																																							
D9252	SD1252	—	本站站错误条 件	存储号码 1 到 16 的 条件	存储主机外的其它发生错误的本站站号码。																																																																																																						
D9253	SD1253	—		存储号码 17 到 32 的条件	<table><tr><th rowspan="2">设备号</th><th colspan="16">位</th></tr><tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr><tr><td>SD1252</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr><tr><td>SD1253</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr><tr><td>SD1254</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr><tr><td>SD1255</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr></table>	设备号	位																b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1252	L16	L15	L14	L13	L12	L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1	SD1253	L32	L31	L30	L29	L28	L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17	SD1254	L48	L47	L46	L45	L44	L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33	SD1255	L64	L63	L62	L61	L60	L59	L58	L57	L56	L55	L54	L53	L52	L51	L50	L49	
设备号	位																																																																																																										
	b15	b14		b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																																										
SD1252	L16	L15	L14	L13	L12	L11	L10	L9	L8	L7	L6	L5	L4	L3	L2	L1																																																																																											
SD1253	L32	L31	L30	L29	L28	L27	L26	L25	L24	L23	L22	L21	L20	L19	L18	L17																																																																																											
SD1254	L48	L47	L46	L45	L44	L43	L42	L41	L40	L39	L38	L37	L36	L35	L34	L33																																																																																											
SD1255	L64	L63	L62	L61	L60	L59	L58	L57	L56	L55	L54	L53	L52	L51	L50	L49																																																																																											
D9254	SD1254	—	存储号码 33 到 48 的条件																																																																																																								
D9255	SD1255	—	存储号码 49 到 64 的条件	和处于错误中的站号码相对应的位变为“1”。 例子：当本站站 12 有错误时，SD1252 中的 b11 变为“1”，并且当 SD1252 被监视时，它的值为“2048(800h)”																																																																																																							

(11) 保险丝烧坏的模块

号码	名称	含义	解释	设定者 (当被设定 时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU																																																																				
SD1300 SD1301 SD1302 SD1303 SD1304 SD1305 SD1306 SD1307 SD1308 SD1309 到 SD1330 SD1331	保险丝烧坏的 模块	以 16 数据点为单元的位模式，表示了保险丝被烧坏的模块。 0: 无烧坏的保险丝 1: 有烧坏的保险丝	<ul style="list-style-type: none">● 保险丝被烧坏的输出模块的数量以位模式输入（以 16 数据点为单元）。（如果模块号码由参数设置，则此参数者设置的号码被存储。）● 对远程站的输出模块也探测保险丝烧坏的情况。 <table><tr><td></td><td>b15</td><td>b14</td><td>b13</td><td>b12</td><td>b11</td><td>b10</td><td>b9</td><td>b8</td><td>b7</td><td>b6</td><td>b5</td><td>b4</td><td>b3</td><td>b2</td><td>b1</td><td>b0</td></tr><tr><td>SD1300</td><td>0</td><td>0</td><td>0</td><td>1 (Y1C0)</td><td>0</td><td>0</td><td>0</td><td>1 (Y80)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>SD1301</td><td>1 (Y1F0)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1 (Y1A)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>SD1331</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1 (Y1F B0)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1 (Y1F 30)</td><td>0</td><td>0</td><td>0</td></tr></table> <p>表示一个烧坏的保险丝</p> <ul style="list-style-type: none">● 即使烧坏的保险丝被新的替换掉，此标志位也不被清除。此标志由错误复位操作清除。		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1300	0	0	0	1 (Y1C0)	0	0	0	1 (Y80)	0	0	0	0	0	0	0	0	SD1301	1 (Y1F0)	0	0	0	0	1 (Y1A)	0	0	0	0	0	0	0	0	0	0	SD1331	0	0	0	0	1 (Y1F B0)	0	0	0	0	0	0	0	1 (Y1F 30)	0	0	0	S(错误)	D9100 D9101 D9102 D9103 D9104 D9105 D9106 D9107 新增 新增	○+Rem
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																										
SD1300	0	0	0	1 (Y1C0)	0	0	0	1 (Y80)	0	0	0	0	0	0	0	0																																																										
SD1301	1 (Y1F0)	0	0	0	0	1 (Y1A)	0	0	0	0	0	0	0	0	0	0																																																										
SD1331	0	0	0	0	1 (Y1F B0)	0	0	0	0	0	0	0	1 (Y1F 30)	0	0	0																																																										
SD1350 到 SD1381	外部供电电源被拆掉的模块 (用于将来的扩展)	以 16 数据点为单元的位模式，表示了外部供电电源被拆掉的模块。 0: 外部供电电源被拆掉 1: 外部供电电源没有被拆掉	外部供电电源被拆掉的模块号码（以 16 数据点为单元）以位模式输入。 (如果模块号码由参数设置，此参数设置的号码被使用。)	S(错误)	新增	QCPU 远程																																																																				

特殊寄存器列表(续表)

(12) I/O 模块校验

号码	名称	含义	解释	设定者 (当被设定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD1400	I/O 模块校验 错误	以 16 数据点为单元的位模式，表示了有校验错误的模块。 0: 无 I/O 校验错误 1: I/O 校验错误存在	<ul style="list-style-type: none">当 I/O 模块，其 I/O 模块信息和系统上电时登记的不一样被检测到时，这些 I/O 模块的号码被以位模式输入。 (如果此 I/O 由参数设置，此参数设置的号码被存储。)也检测 I/O 模块信息。 <div><div>D9116</div><div>D9117</div><div>D9123</div><div>b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0</div><div>0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1</div><div>0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0</div><div>0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0</div><div>(¹/₁₆) (¹/₁₆) (¹/₁₆)</div><div>表示一个 I/O 模块校验错误</div></div> <ul style="list-style-type: none">即使烧坏的保险丝被新的替换掉，此标志位也不被清除。 此标志由错误复位操作清除。	S(错误)	D9116	○+Rem
SD1401					D9117	
SD1402					D9118	
SD1403					D9119	
SD1404					D9120	
SD1405					D9121	
SD1406					D9122	
SD1407					D9123	
SD1408					新增	
SD1409					新增	
SD1430					新增	
SD1431					新增	

(13) 用于冗余系统(主机系统 CPU 信息 *1)，只用于 Q4AR

SD1510 至 SD1599 只对冗余系统有效。对于单个系统它们都被设为 0。

号码	名称	含义	解释	设定者 (当被设定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD1500	基本周期	基本周期	<ul style="list-style-type: none">设置用于使用浮点数数据的过程控制指令的基本周期(1 秒为单位) <div>浮点数数据 = <div>SD1501</div><div>SD1500</div></div>	U	新增	Q4AR
SD1501						
SD1502	过程控制指令详细错误代码	过程控制指令详细错误代码	<ul style="list-style-type: none">表示了发生在过程控制指令中的错误的详细错误内容。	S (发生错误)	新增	
SD1503	过程控制指令产生的错误的定位	过程控制指令产生的错误的定位	<ul style="list-style-type: none">表示了过程控制指令中发生错误的错误过程块	S (发生错误)	新增	
SD1512	CPU 模块启动过程中的操作模式	热启动切换电源断开时间	<ul style="list-style-type: none">表示了 CPU 模块启动时，在操作模式中，从热启动到初始化启动的切换过程中，电源断开的的时间(S)	S(初始化)	新增	
SD1590	切换请求网络号码	请求起源网络号码	<ul style="list-style-type: none">存储当 SM1590 变为 ON 时的请求起源网络号码。	S (发生错误)	新增	

*1 存储主机系统 CPU 信息。

(14) 用于冗余系统 (其它系统 CPU 信息*1)，只用于 Q4AR

SD1600 至 SD1659 只在用于冗余系统的备份模式中有效，并且在分离模式中刷新无法被执行。
在单独系统中 SD1600 至 SD1699 都为 0。

特殊寄存器列表

号码	名称	含义	解释	设定者 (当被设定时)	相应的 ACPU SD <table><tr><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr></table> * 2	<input type="checkbox"/>	<input type="checkbox"/>	相应的 CPU
<input type="checkbox"/>	<input type="checkbox"/>							
SD1600	诊断错误	诊断错误号码	<ul style="list-style-type: none">以 BIN 码存储在其系统 CPU 模块诊断过程中发生的错误的错误号码。存储当前发生的最新错误。	S (每个 END)	新增	Q4AR		
SD1601	诊断错误发生 时间	诊断错误发生时间	<ul style="list-style-type: none">SD1600 存储更新的日期和时间。存储每个 BCD 的两位数字。对于存储状态参考 SD1 到 SD3 (SD1→SD1601, SD2→SD1602, SD3→SD1603)	S (每个 END)	新增			
SD1602								
SD1603								
SD1604	错误信息分类	错误信息分类	<ul style="list-style-type: none">存储错误公用信息 / 单独信息的分类代码。对于存储状态参考 SD4。	S (每个 END)	新增			
SD1605	错误公用信息	错误公用信息	<ul style="list-style-type: none">存储错误代码的公用信息。对于存储状态参考 SD5 到 SD15。 (SD5→SD1605, SD6→SD1606, SD7→SD1607, SD8→SD1608, SD9→SD1609, SD10→SD1610, SD11→SD1611, SD12→SD1612, SD13→SD1613, SD14→SD1614, SD15→SD1615)	S (每个 END)	新增			
SD1606								
SD1607								
SD1608								
SD1609								
SD1610								
SD1611								
SD1612								
SD1613								
SD1614								
SD1615								
SD1616	错误单独信息	错误单独信息	<ul style="list-style-type: none">存储错误代码的公用信息。对于存储状态参考 SD16 到 SD26。 (SD16→SD1616, SD17→SD1617, SD18→SD1618, SD19→SD1619, SD20→SD1620, SD21→SD1621, SD22→SD1622, SD23→SD1623, SD24→SD1624, SD25→SD1625, SD26→SD1626)	S (每个 END)	新增			
SD1617								
SD1618								
SD1619								
SD1620								
SD1621								
SD1622								
SD1623								
SD1624								
SD1625								
SD1626								
SD1650	开关状态	CPU 模块开关状态	<ul style="list-style-type: none">存储 CPU 模块的开关状态。存储状态参考 SD200。(SD1650→SD200)	S (每个 END)	新增			
SD1651	LED 状态	CPU 模块-LED 状态	<ul style="list-style-type: none">存储 CPU 模块的 LED 状态。关闭时为 0，打开时为 1，闪烁时为 2。存储状态参考 SD201。(SD1651→SD201)	S (每个 END)	新增			
SD1653	CPU 模块操作状态	CPU 模块操作状态	<ul style="list-style-type: none">存储 CPU 模块操作状态。存储状态参考 SD203。 (SD1653→SD203)	S (每个 END)	新增			

* 1 存储其它系统 CPU 模块自诊断信息和系统信息。

* 2 表示用于主机系统 CPU 模块的特殊寄存器 (SD ☐ ☐)。

(15) 用于冗余系统 (跟踪)，只用于 Q4AR

SD1700 至 SD1799 只对冗余系统有效。
对于独立系统这些都为 0。

号码	名称	含义	解释	解释 (当被设定时)	相应的 ACPU SD <input type="checkbox"/> <input type="checkbox"/> * 2	相应的 CPU
SD1700	跟踪错误检测计数	跟踪错误检测计数	<ul style="list-style-type: none"> 将跟踪错误检测到的数量+1。 	发生错误	新增	Q4AR

* 2 表示用于主机系统 CPU 模块的特殊寄存器 (SD ☐ ☐)。

附录 4.3 过程控制 CPU 的特殊寄存器列表

特殊寄存器 SD 是应用固化在 PLC 中的内部寄存器。
由于这个原因，这些寄存器不能以普通的寄存器那样的方式在顺控程序中被使用。
然而，可以按照需要写入数据以控制 CPU 模块和远程 I/O 模块。
如果没有特殊的相反指定，数据以 BIN 值的形式存储在特殊寄存器内。

下表中标题的含义如下所示。

项目	项目的功能
号码	● 表示特殊寄存器的号码
名称	● 表示特殊寄存器的名称
含义	● 表示特殊寄存器的内容
解释	● 讨论特殊寄存器内容的详细信息
设置者(当被设置时)	● 表示寄存器是被系统设置还是被用户设置，并且，如果是被系统设置，当设置被执行时。 <设置者> S : 由系统设置 U : 由用户设置(从 GX Developer 或类似设备上的顺序程序或者测试操作中) S/U : 由系统和用户设置 <当被设置时> → 只有当寄存器由系统设置时才被指示 每个 END : 在每条 END 处理过程中被设置 初始化 : 在初始化处理时被设置(当系统电源变为 ON, 或者当从 STOP 变成 RUN 时) 状态改变 : 只有在状态有改变时才被设置 错误 : 只有在错误产生时才被设置 指令执行 : 在指令被执行时被设置 请求 : 只有在有用户请求时才被设置(通过 SM, 等)
相应的 ACPU M9 □ □ □	● 表示 ACPU(D9 □ □ □)中相应的特殊寄存器(当有内容发生变化时的变化和符号) ● “新增”表示此项目是被新增加到过程控制 CPU 中
相应的 CPU	● 表示相应的 CPU 模块类型名称 ○+Rem: 可以被应用到过程控制 CPU 和 MELSECNET/H 远程 I/O 模块 ○: 可以被应用到所有类型的 CPU 模块 远程: 可以被应用到 MELSECNET/H 远程 I/O 模块

- 对于下列项目的详细信息，参见下列手册：：
- 网络 → ● 用于和 Q 对应的 MELSECNET/H 网络系统参考手册(PLC 至 PLC 网络)
● 用于和 Q 对应的 MELSECNET/H 网络系统参考手册(远程 I/O 网络)
 - SFC → QCPU(Q 模式)/QnACPU 编程手册(SFC)

要点	
SD1200 到 SD1255 用于 QnACPU。 这些继电器在过程控制 QCPU 中为空。	

特殊寄存器列表

(1) 诊断信息

号码	名称	含义	解释	设定者 (当被设定 时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU						
SD0	诊断错误	诊断错误代码	<ul style="list-style-type: none">● 诊断发现的错误的错误代码以 BIN 数据形式存储。● 内容和最新的故障历史信息相同。	S (错误)	D9008 格式改变							
SD1	诊断错误发生的 时钟时间	诊断错误发生的 时钟时间	<ul style="list-style-type: none">● SD0 数据被更新时的年(最后两位数)和月以 BCD 2 位码形式存储。 <div>(例子)</div><div><table><tr><td>b15 到 b8</td><td>b7 到 b0</td><td></td></tr><tr><td>年(0到99)</td><td>月(1到12)</td><td>:十月, 1995 9510H</td></tr></table></div>	b15 到 b8	b7 到 b0		年(0到99)	月(1到12)	:十月, 1995 9510H	S(错误)	新增	
b15 到 b8			b7 到 b0									
年(0到99)			月(1到12)	:十月, 1995 9510H								
SD2	<ul style="list-style-type: none">● SD0 数据被更新时的日期和小时以 BCD 2 位码形式存储。 <div>(例子)</div><div><table><tr><td>b15 到 b8</td><td>b7 到 b0</td><td></td></tr><tr><td>日期(1到31)</td><td>小时(0到23)</td><td>:10 p.m. 在25号 2510H</td></tr></table></div>	b15 到 b8	b7 到 b0		日期(1到31)	小时(0到23)	:10 p.m. 在25号 2510H					
b15 到 b8	b7 到 b0											
日期(1到31)	小时(0到23)	:10 p.m. 在25号 2510H										
SD3	<ul style="list-style-type: none">● SD0 数据被更新时的分钟和秒以 BCD 2 位码的形式存储。 <div>(例子)</div><div><table><tr><td>b15 到 b8</td><td>b7 到 b0</td><td></td></tr><tr><td>分钟(0到59)</td><td>秒(0到59)</td><td>:35分48秒 (在几点后又过了) 3548H</td></tr></table></div>	b15 到 b8	b7 到 b0		分钟(0到59)	秒(0到59)	:35分48秒 (在几点后又过了) 3548H					
b15 到 b8	b7 到 b0											
分钟(0到59)	秒(0到59)	:35分48秒 (在几点后又过了) 3548H										
SD4	错误信息分类	错误信息分类 代码	<ul style="list-style-type: none">● 用于标志存储在公用信息区域(SD5 到 SD15)和单独信息区域(SD16 到 SD26)的错误信息是哪种类型的分类代码存储在这里。 <div><table><tr><td>b15 到 b8</td><td>b7 到 b0</td><td></td></tr><tr><td>单独信息种类代码</td><td>公用信息种类代码</td><td></td></tr></table></div>● 公用信息种类代码存储了下列一些代码： 0：无错误 1：单元/模块号码/ PLC 号码/基板号码 * 2：文件名/驱动器名 3：时间(设定值) 4：程序错误定位 5：切换原因(只用于 Q4AR) *：对于一个多 PLC 系统，模块号或 PLC 号码的存储依据于发生的错误。 (参见存储的每个号码对应的错误代码。) 1 号 PLC：1, 2 号 PLC：2, 3 号 PLC 3, 4 号 PLC：4● 单个信息分类代码存储了下列一些代码： 0：无错误 1：(打开) 2：文件名/驱动器名 3：时间(实际测量值) 4：程序错误定位 5：参数号 6：报警器号 7：检查指令故障号码	b15 到 b8	b7 到 b0		单独信息种类代码	公用信息种类代码		S(错误)	新增	○+Rem
b15 到 b8	b7 到 b0											
单独信息种类代码	公用信息种类代码											

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU																																					
SD5	错误公用信息	错误公用信息	<ul style="list-style-type: none">● 和错误代码(SD0)相应的公用信息存储在这里。● 接下来的四种类型的信息存储在这里：<ul style="list-style-type: none">① 插槽号 <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD5</td><td>插槽号/基板号码*1*2</td></tr><tr><td>SD6</td><td>I/O号*3</td></tr><tr><td>SD7</td><td rowspan="9">(空白)</td></tr><tr><td>SD8</td></tr><tr><td>SD9</td></tr><tr><td>SD10</td></tr><tr><td>SD11</td></tr><tr><td>SD12</td></tr><tr><td>SD13</td></tr><tr><td>SD14</td></tr><tr><td>SD15</td></tr></table> <p>(不用作基板号码)</p> <p>*1: 在多 CPU 系统中, 插槽号或者 CPU 号依据发生的错误被保存。 多 CPU 系统中的插槽 0 指的是最右边的 CPU 模块右边的插槽。 (参见用于哪个号码被存储的错误代码) 第 1 CPU: 1, 第 2 CPU: 2, 第 3 CPU: 3, 第 4 CPU: 4</p> <p>*2: 如果 MELSECNET/H 远程 I/O 站点中发生保险丝烧坏或者 I/O 校验错误, 网络号码被存储到高 8 位, 站点号码被存储到低 8 位。 使用 I/O 号码检查发生保险丝烧坏或者 I/O 校验错误的模块。</p> <ul style="list-style-type: none">② 文件名/驱动器名 <p>(例子)</p> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD5</td><td>驱动器</td></tr><tr><td>SD6</td><td rowspan="5">文件名 (ASCII码:8字符)</td></tr><tr><td>SD7</td></tr><tr><td>SD8</td></tr><tr><td>SD9</td></tr><tr><td>SD10</td></tr><tr><td>SD10</td><td>扩展*3</td><td>2EH(.)</td></tr><tr><td>SD11</td><td colspan="2">(ASCII码:3字符)</td></tr><tr><td>SD12</td><td rowspan="4">(空白)</td></tr><tr><td>SD13</td></tr><tr><td>SD14</td></tr><tr><td>SD15</td></tr></table> <p>文件名= ABCDEFGH. IJK b15 到 b8 b7 到 b0 42H(B) 41H(A) 44H(D) 43H(C) 46H(F) 45H(E) 48H(H) 47H(G) 49H(I) 2EH(.) 4BH(K) 4AH(J)</p>	号码	含义	SD5	插槽号/基板号码*1*2	SD6	I/O号*3	SD7	(空白)	SD8	SD9	SD10	SD11	SD12	SD13	SD14	SD15	号码	含义	SD5	驱动器	SD6	文件名 (ASCII码:8字符)	SD7	SD8	SD9	SD10	SD10	扩展*3	2EH(.)	SD11	(ASCII码:3字符)		SD12	(空白)	SD13	SD14	SD15	S(错误)	新增	○+Rem
号码			含义																																								
SD5			插槽号/基板号码*1*2																																								
SD6			I/O号*3																																								
SD7			(空白)																																								
SD8																																											
SD9																																											
SD10																																											
SD11																																											
SD12																																											
SD13																																											
SD14																																											
SD15																																											
号码			含义																																								
SD5			驱动器																																								
SD6	文件名 (ASCII码:8字符)																																										
SD7																																											
SD8																																											
SD9																																											
SD10																																											
SD10	扩展*3	2EH(.)																																									
SD11	(ASCII码:3字符)																																										
SD12	(空白)																																										
SD13																																											
SD14																																											
SD15																																											
SD6																																											
SD7																																											
SD8																																											
SD9																																											
SD10																																											
SD11																																											
SD12																																											
SD13																																											
SD14																																											
SD15																																											

*3: 参考备注

备注

1) 扩展型如下所示

SD10 高 8 位	SD11		扩展名称	文件类型
	低 8 位	高 8 位		
51H	50H	41H	QPA	参数
51H	50H	47H	QPG	顺序程序/SFC 程序
51H	43H	44H	QCD	设备注释
51H	44H	49H	QDI	设备初始值
51H	44H	52H	QDR	文件寄存器
51H	44H	53H	QDS	仿真数据
51H	44H	4CH	QDL	本地设备
51H	46H	44H	QFD	故障历史数据

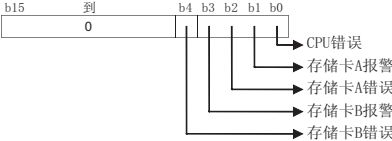
特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU																																																							
SD5	错误公用信息	错误公用信息	<div>③ 时间(设定值)</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD5</td><td>时间:1μs单位(0到999μs)</td></tr><tr><td>SD6</td><td>时间:1ms单位(0到65535ms)</td></tr><tr><td>SD7</td><td rowspan="9">(空白)</td></tr><tr><td>SD8</td></tr><tr><td>SD9</td></tr><tr><td>SD10</td></tr><tr><td>SD11</td></tr><tr><td>SD12</td></tr><tr><td>SD13</td></tr><tr><td>SD14</td></tr><tr><td>SD15</td></tr></table> <div>④ 程序错误定位</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD5</td><td rowspan="4">文件名 (ASCII码:8字符)</td></tr><tr><td>SD6</td></tr><tr><td>SD7</td></tr><tr><td>SD8</td></tr><tr><td>SD9</td><td>扩展</td></tr><tr><td>SD10</td><td>2EH(.) (ASCII码:3字符)</td></tr><tr><td>SD11</td><td>模式*4</td></tr><tr><td>SD12</td><td>块号码</td></tr><tr><td>SD13</td><td>步号码/转换号码</td></tr><tr><td>SD14</td><td>顺序步号码(L)</td></tr><tr><td>SD15</td><td>顺序步号码(H)</td></tr></table> <div>* 4 模式数据中的内容</div> <table><tr><td>15</td><td>14</td><td>到</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>← (位号码)</td></tr><tr><td>0</td><td>0</td><td>到</td><td>0</td><td>0</td><td>*</td><td>*</td><td>*</td><td></td></tr></table> <div>(未被使用)</div> <div>SFC块指定存在(1)/不存在(0)</div> <div>SFC步指定存在(1)/不存在(0)</div> <div>SFC转换指定存在(1)/不存在(0)</div>	号码	含义	SD5	时间:1μs单位(0到999μs)	SD6	时间:1ms单位(0到65535ms)	SD7	(空白)	SD8	SD9	SD10	SD11	SD12	SD13	SD14	SD15	号码	含义	SD5	文件名 (ASCII码:8字符)	SD6	SD7	SD8	SD9	扩展	SD10	2EH(.) (ASCII码:3字符)	SD11	模式*4	SD12	块号码	SD13	步号码/转换号码	SD14	顺序步号码(L)	SD15	顺序步号码(H)	15	14	到	4	3	2	1	0	← (位号码)	0	0	到	0	0	*	*	*		S(错误)	新增	○+Rem
号码				含义																																																									
SD5				时间:1μs单位(0到999μs)																																																									
SD6				时间:1ms单位(0到65535ms)																																																									
SD7				(空白)																																																									
SD8																																																													
SD9																																																													
SD10																																																													
SD11																																																													
SD12																																																													
SD13																																																													
SD14																																																													
SD15																																																													
号码				含义																																																									
SD5				文件名 (ASCII码:8字符)																																																									
SD6																																																													
SD7																																																													
SD8																																																													
SD9	扩展																																																												
SD10	2EH(.) (ASCII码:3字符)																																																												
SD11	模式*4																																																												
SD12	块号码																																																												
SD13	步号码/转换号码																																																												
SD14	顺序步号码(L)																																																												
SD15	顺序步号码(H)																																																												
15	14	到	4	3	2	1	0	← (位号码)																																																					
0	0	到	0	0	*	*	*																																																						
SD6																																																													
SD7																																																													
SD8																																																													
SD9																																																													
SD10																																																													
SD11																																																													
SD12																																																													
SD13																																																													
SD14																																																													
SD15																																																													
SD15																																																													
SD15																																																													
SD15																																																													

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定 时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU																																																																																																																									
SD16	错误单独信息	错误单独信息	<div>● 和错误代码(SD0)相应的单独信息存储在此。</div> <div>① 文件名/驱动器名</div> <div>(例子)</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td>驱动器</td></tr><tr><td>SD17</td><td rowspan="4">文件名 (ASCII码:8字符)</td></tr><tr><td>SD18</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td></tr><tr><td>SD21</td><td>扩展</td><td>2EH(.)</td></tr><tr><td>SD22</td><td>(ASCII码:3字符)</td></tr><tr><td>SD23</td><td rowspan="4">(空白)</td></tr><tr><td>SD24</td></tr><tr><td>SD25</td></tr><tr><td>SD26</td></tr></table> <div>② 时间(实际测量值)</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td>时间:1μs单位(0到999μs)</td></tr><tr><td>SD17</td><td>时间:1ms单元(0到65535ms)</td></tr><tr><td>SD18</td><td rowspan="9">(空白)</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td></tr><tr><td>SD21</td></tr><tr><td>SD22</td></tr><tr><td>SD23</td></tr><tr><td>SD24</td></tr><tr><td>SD25</td></tr><tr><td>SD26</td></tr></table> <div>③ 程序错误定位</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td rowspan="4">文件名 (ASCII码:8字符)</td></tr><tr><td>SD17</td></tr><tr><td>SD18</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td><td>扩展</td><td>2EH(.)</td></tr><tr><td>SD21</td><td>(ASCII码:3字符)</td></tr><tr><td>SD22</td><td>模式*</td></tr><tr><td>SD23</td><td>块号码</td></tr><tr><td>SD24</td><td>步号码/转换号码</td></tr><tr><td>SD25</td><td>顺序步号码(L)</td></tr><tr><td>SD26</td><td>顺序步号码(H)</td></tr></table> <div>* 模式数据中的内容</div> <table><tr><td>15</td><td>14</td><td>到</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>←(位号码)</td></tr><tr><td>0</td><td>0</td><td>到</td><td>0</td><td>0</td><td>*</td><td>*</td><td>*</td><td></td></tr></table> <div>(未使用)</div> <div>SFC块指定存在(1)/不存在(0)</div> <div>SFC步指定存在(1)/不存在(0)</div> <div>SFC转换指定存在(1)/不存在(0)</div> <div>④ 参数号码</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td>参数号码*6</td></tr><tr><td>SD17</td><td rowspan="10">(空白)</td></tr><tr><td>SD18</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td></tr><tr><td>SD21</td></tr><tr><td>SD22</td></tr><tr><td>SD23</td></tr><tr><td>SD24</td></tr><tr><td>SD25</td></tr><tr><td>SD26</td></tr></table> <div>⑤ 报警器号码/CHK 指</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td>号码</td></tr><tr><td>SD17</td><td rowspan="10">(空白)</td></tr><tr><td>SD18</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td></tr><tr><td>SD21</td></tr><tr><td>SD22</td></tr><tr><td>SD23</td></tr><tr><td>SD24</td></tr><tr><td>SD25</td></tr><tr><td>SD26</td></tr></table> <div>⑥ 智能功能模块参数 错误 (只用于 QCPU)</div> <table><tr><th>号码</th><th>含义</th></tr><tr><td>SD16</td><td>参数号码*6</td></tr><tr><td>SD17</td><td>用于智能功能模块的错 误代码</td></tr><tr><td>SD18</td><td rowspan="19">(空白)</td></tr><tr><td>SD19</td></tr><tr><td>SD20</td></tr><tr><td>SD21</td></tr><tr><td>SD22</td></tr><tr><td>SD23</td></tr><tr><td>SD24</td></tr><tr><td>SD25</td></tr><tr><td>SD26</td></tr></table> <div>* 6 对于参数号码的详细信息, 参考使用的 CPU 模块的用户手 册(功能解释, 编程基础)。</div>	号码	含义	SD16	驱动器	SD17	文件名 (ASCII码:8字符)	SD18	SD19	SD20	SD21	扩展	2EH(.)	SD22	(ASCII码:3字符)	SD23	(空白)	SD24	SD25	SD26	号码	含义	SD16	时间:1μs单位(0到999μs)	SD17	时间:1ms单元(0到65535ms)	SD18	(空白)	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26	号码	含义	SD16	文件名 (ASCII码:8字符)	SD17	SD18	SD19	SD20	扩展	2EH(.)	SD21	(ASCII码:3字符)	SD22	模式*	SD23	块号码	SD24	步号码/转换号码	SD25	顺序步号码(L)	SD26	顺序步号码(H)	15	14	到	4	3	2	1	0	←(位号码)	0	0	到	0	0	*	*	*		号码	含义	SD16	参数号码*6	SD17	(空白)	SD18	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26	号码	含义	SD16	号码	SD17	(空白)	SD18	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26	号码	含义	SD16	参数号码*6	SD17	用于智能功能模块的错 误代码	SD18	(空白)	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26	S(错误)	新增	○+Rem
号码			含义																																																																																																																												
SD16			驱动器																																																																																																																												
SD17			文件名 (ASCII码:8字符)																																																																																																																												
SD18																																																																																																																															
SD19																																																																																																																															
SD20																																																																																																																															
SD21			扩展	2EH(.)																																																																																																																											
SD22			(ASCII码:3字符)																																																																																																																												
SD23			(空白)																																																																																																																												
SD24																																																																																																																															
SD25																																																																																																																															
SD26																																																																																																																															
号码			含义																																																																																																																												
SD16			时间:1μs单位(0到999μs)																																																																																																																												
SD17			时间:1ms单元(0到65535ms)																																																																																																																												
SD18			(空白)																																																																																																																												
SD19																																																																																																																															
SD20																																																																																																																															
SD21																																																																																																																															
SD22																																																																																																																															
SD23																																																																																																																															
SD24																																																																																																																															
SD25																																																																																																																															
SD26																																																																																																																															
号码			含义																																																																																																																												
SD16	文件名 (ASCII码:8字符)																																																																																																																														
SD17																																																																																																																															
SD18																																																																																																																															
SD19																																																																																																																															
SD20	扩展	2EH(.)																																																																																																																													
SD21	(ASCII码:3字符)																																																																																																																														
SD22	模式*																																																																																																																														
SD23	块号码																																																																																																																														
SD24	步号码/转换号码																																																																																																																														
SD25	顺序步号码(L)																																																																																																																														
SD26	顺序步号码(H)																																																																																																																														
15	14	到	4	3	2	1	0	←(位号码)																																																																																																																							
0	0	到	0	0	*	*	*																																																																																																																								
号码	含义																																																																																																																														
SD16	参数号码*6																																																																																																																														
SD17	(空白)																																																																																																																														
SD18																																																																																																																															
SD19																																																																																																																															
SD20																																																																																																																															
SD21																																																																																																																															
SD22																																																																																																																															
SD23																																																																																																																															
SD24																																																																																																																															
SD25																																																																																																																															
SD26																																																																																																																															
号码	含义																																																																																																																														
SD16	号码																																																																																																																														
SD17	(空白)																																																																																																																														
SD18																																																																																																																															
SD19																																																																																																																															
SD20																																																																																																																															
SD21																																																																																																																															
SD22																																																																																																																															
SD23																																																																																																																															
SD24																																																																																																																															
SD25																																																																																																																															
SD26																																																																																																																															
号码	含义																																																																																																																														
SD16	参数号码*6																																																																																																																														
SD17	用于智能功能模块的错 误代码																																																																																																																														
SD18	(空白)																																																																																																																														
SD19																																																																																																																															
SD20																																																																																																																															
SD21																																																																																																																															
SD22																																																																																																																															
SD23																																																																																																																															
SD24																																																																																																																															
SD25																																																																																																																															
SD26																																																																																																																															

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD50	错误复位	执行错误复位的错误号码	<ul style="list-style-type: none"> ● 存储执行错误复位的错误号码 	U	新增	○+Rem
SD51	电池低闭锁	位模式表示了哪里发生了电池电压下降	<ul style="list-style-type: none"> ● 当电池电压下降时所有的相应位变为 1 (ON)。 ● 随后, 即使此后电池电压恢复为正常这些位也保持为 1 (ON)。  <ul style="list-style-type: none"> ● 当电池电压下降发生时报警数据可以在指定的时间内被保持。 ● 此错误表示电池充电的完成。 ● 当 QCPU 被使用时, 存储卡 B 为标准的, 并且因此相应位保持常 OFF。 	S (错误)	新增	○
SD52	电池低	位模式表示了哪里发生了电池电压下降	<ul style="list-style-type: none"> ● 和上面 SD51 的组态一样 ● 变为 0 (OFF), 当电池电压此后恢复为正常。 ● 当过程控制 CPU 被使用时, 存储卡 B 为标准的, 并且因此相应位保持常 OFF。 	S (错误)	新增	
SD53	AC DOWN 检测	AC DOWN 的次数	<ul style="list-style-type: none"> ● 在 CPU 模块的计算过程中, 每次输入电压下降到或低于额定电压的 85% (AC 电源) / 65% (DC 电源), 此值增加 1 并被以 BIN 码存储。 	S (错误)	D9005	○+Rem
SD60	烧坏的保险丝号码	烧坏保险丝的模块的数量	<ul style="list-style-type: none"> ● 存储在这里的值, 是最低的有保险丝烧坏模块的站点 I/O 号码。 	S (错误)	D9000	○+Rem
SD61	I/O 模块校验错误号码	I/O 模块校验错误模块号码	<ul style="list-style-type: none"> ● I/O 模块校验号码发生处的模块最低 I/O 号码。 	S (错误)	D9002	
SD62	报警器号码	报警器号码	<ul style="list-style-type: none"> ● 被探测到的第一个报警器号码 (F 号码) 存储在这里。 	S (指令执行)	D9009	○
SD63	报警器的数量	报警器的数量	<ul style="list-style-type: none"> ● 存储搜索到的报警器的数量。 	S (指令执行)	D9124	

特殊寄存器列表(续表)

号码	名称	含义	解释	设定者 (当被设定 时)	相应的 ACPU D9 <div><div></div><div></div><div></div></div>	相应的 CPU				
SD64	探测到的报警 器号码列表	探测到的报警 器号码	<div>当 F 由于 <div>OUT F</div> 或 <div>SET F</div> 变为 ON 时, 从 SD64 到 SD79 逐次变为 ON 的 F 号码被登记。</div> <div>被 <div>RST F</div> 变为 OFF 的 F 号码从 SD64 到 SD79 中删除, 并且在删除的 F 号码移位到前面的寄存器后, 这些 F 号码被存储。</div> <div><div>LEDR</div> 指令的执行使 SD64 到 SD79 中的内容向上移动一位。</div> <div>(这也可以通过 Q3A/Q4ACPU 前面的 INDICATOR RESET 开关来实现。)</div> <div>在 16 个报警器被探测到后, 第 17 个探测并不被保存到 SD64 到 SD79 中。</div> <div><div>SET SET SET SET SET SET SET SET SET SET SET SET SET SET SET SET</div><div>F50 F25 F99 F25 F15 F70 F65 F38 F110F151F210 LEDR</div><div><div><div>SD62</div><div><div>0</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>99</div></div><div>... (探测到的号码)</div></div><div><div>SD63</div><div><div>0</div><div>1</div><div>2</div><div>3</div><div>2</div><div>3</div><div>4</div><div>5</div><div>6</div><div>7</div><div>8</div><div>9</div><div>8</div></div><div>... (探测到的报警器的数量)</div></div><div><div><div><div>SD64</div><div><div>0</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>50</div><div>99</div></div></div><div><div>SD65</div><div><div>0</div><div>0</div><div>25</div><div>25</div><div>99</div><div>99</div><div>99</div><div>99</div><div>99</div><div>99</div><div>99</div><div>99</div><div>15</div></div></div><div><div>SD66</div><div><div>0</div><div>0</div><div>0</div><div>99</div><div>0</div><div>15</div><div>15</div><div>15</div><div>15</div><div>15</div><div>15</div><div>15</div><div>70</div></div></div><div><div>SD67</div><div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>70</div><div>70</div><div>70</div><div>70</div><div>70</div><div>65</div></div></div><div><div>SD68</div><div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>65</div><div>65</div><div>65</div><div>65</div><div>65</div><div>38</div></div></div><div><div>SD69</div><div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>38</div><div>38</div><div>38</div><div>38</div><div>110</div></div></div><div><div>SD70</div><div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>110</div><div>110</div><div>110</div><div>151</div></div></div><div><div>SD71</div><div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>151</div><div>151</div><div>210</div></div></div><div><div>SD72</div><div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>210</div><div>0</div></div></div><div><div>SD73</div><div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div></div><div><div>SD74</div><div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div></div><div><div>SD75</div><div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div></div><div><div>SD76</div><div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div></div><div><div>SD77</div><div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div></div><div><div>SD78</div><div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div></div><div><div>SD79</div><div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div></div></div><div>(探测到的号码)</div></div></div><div>S (指令执行)</div><div>D9125</div><div>D9126</div><div>D9127</div><div>D9128</div><div>D9129</div><div>D9130</div><div>D9131</div><div>D9132</div><div>新增</div><div>新增</div><div>新增</div><div>新增</div><div>新增</div><div>新增</div><div>新增</div><div>新增</div></div>							
SD80				CHK 号码	CHK 号码	● 由 CHK 指令探测到的错误代码被存储成 BCD 码格式。	S (指令执行)	新增		
SD90				步转换看门狗 定时器设置值 (只有在 SFC 程序存在时才被使能)	用于定时器设置值和时间用 完错误的 F 号码	对应于 SM90	<div>● 设置报警器号码 (F 号码), 此报警器在步转换看门狗定时器设置或者看门狗定时器时间界限错误发生时变为 ON。</div> <div><div>b15 到 b8</div><div>b7 到 b0</div><div><div></div><div></div></div><div>F 号码设置 (0到255)</div><div>定时器时间界限设置 (1到255s: (1s为单位))</div></div> <div>● 在活动步启动此定时器的过程中将 SM90 到 SM99 中的任何一个变为 ON, 并且如果对应步下面的转换条件在定时器的时间界限内没有被满足, 则设置的报警器 (F) 变为 ON。</div>	U	D9108	
SD91						对应于 SM91			D9109	
SD92						对应于 SM92			D9110	
SD93						对应于 SM93			D9111	
SD94						对应于 SM94			D9112	
SD95						对应于 SM95			D9113	
SD96						对应于 SM96			D9114	
SD97						对应于 SM97			新增	
SD98						对应于 SM98			新增	
SD99						对应于 SM99			新增	
SD105	CH1 传输速度 设置(RS232)	当使用 GX Developer 时, 存储预设的传输速度	3 :300bps, 6 :600bps, 24 :2400bps, 48:4800bps 96 :9600bps, 192 :19.2kbps, 384 :38.4kbps 576 :57.6kbps, 1152 :115.2kbps	S	新增	○ +Rem				

特殊寄存器列表

(2) 系统信息

号码	名称	含义	解释	设定者 (当被设定 定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD200	开关状态	CPU 开关的状态	<ul style="list-style-type: none">● 远程 I/O 模块的开关状态被存储成下列格式。 	S(一直)	新增	远程
			<ul style="list-style-type: none">① 远程 I/O 模块开关状态 常 1: STOP● CPU 模块开关状态被存储成下列格式:  <div><div>①: CPU 开关状态</div><div>0: RUN 1: STOP 2: L. CLR</div><div>②: 存储卡开关</div><div>常 OFF</div><div>③: DIP 开关</div><div>b8 到 b12 对应于系统设置开关 1 的 SW1 到 SW5 0: OFF, 1: ON b13 到 b15 为空白</div></div>	S(每个 END 处理)	新增	○
SD201	LED 状态	CPU-LED 的状态	<ul style="list-style-type: none">● 下面的位模式用于存储 CPU 模块上 LED 的状态:  <div><div>①: RUN</div><div>②: ERR.</div><div>③: USER</div><div>④: BAT. ALARM</div><div>⑤: BOOT</div><div>⑥: 空</div><div>⑦: 空</div><div>⑧: MODE</div></div> <div>用于 MODE 的的位模式 0: OFF, 1: 绿色</div>	S(状态改变)	新增	QCPU
SD203	CPU 的工作状态	CPU 的工作状态	<ul style="list-style-type: none">● 远程 I/O 模块的工作状态被存储成下列格式。 	S(常)	新增	Rem
			<ul style="list-style-type: none">① 远程 I/O 模块的工作状态 常 2: STOP● CPU 模块的工作状态按照下图中所示被存储:  <div><div>①: CPU 的工作状态</div><div>0 :RUN 1 :STEP-RUN 2 :STOP 3 :PAUSE</div><div>②: STOP/PAUSE 原因</div><div>0 :开关 1 :远程触点 2 :来自 GX Developer 或串行通信的远程操作 3 :内部程序指令</div><div>注意: 优先级为最早的最高</div><div>4 :错误</div></div>	S(每个 END 处理)	D9015 格式改变	○

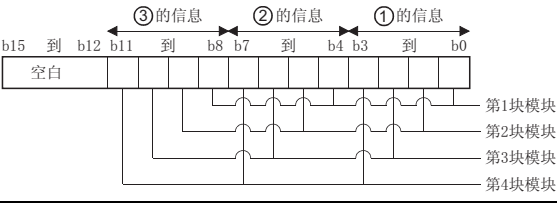
特殊寄存器列表(续表)

号码	名称	含义	解释	设定者 (当被设定 定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU																															
SD206	设备测试执行类型	0: 测试未被执行 1: 处于 X 设备测试中 2: 处于 Y 设备测试中 3: 处于 X/Y 设备测试中	● 当设备测试在 GX Developer 上执行时被设置。	S(请求)	新增	Rem																															
SD207	LED 显示优先级	优先级 1 到 4	● 当产生错误时, LED 将对应于错误号码设置优先级进行显示(闪烁)。 ● 优先级的设置区域如下所示: <div><div>b15 到 b12b11 到 b8b7 到 b4b3 到 b0</div><table><tr><td>SD207</td><td>优先级4</td><td>优先级3</td><td>优先级2</td><td>优先级1</td></tr><tr><td>SD208</td><td>优先级8</td><td>优先级7</td><td>优先级6</td><td>优先级5</td></tr><tr><td>SD209</td><td></td><td></td><td>优先级10</td><td>优先级9</td></tr></table><div>默认值SD207=4321H SD208=8765H SD209=00A9H</div></div> ● 如果“0”被设置了, 则没有显示。 然而, 即使“0”被设置了, 关于 CPU 工作停止(包括参数设置)错误的信息依然会被 LED 无条件地指示。 关于优先级的次序, 参见章节 7.9.5 的备注	SD207	优先级4	优先级3	优先级2	优先级1	SD208	优先级8	优先级7	优先级6	优先级5	SD209			优先级10	优先级9	U	D9038	○																
SD207		优先级4		优先级3	优先级2	优先级1																															
SD208		优先级8		优先级7	优先级6	优先级5																															
SD209			优先级10	优先级9																																	
SD208	优先级 5 到 8	D3039 格式改变																																			
SD209	优先级 9 到 10	新增																																			
SD210	时钟数据	时钟数据 (年, 月)	● 年(后两位数)和月如下所示以 BCD 码存储在 SD210 中: <div><div>b15 到 b12b11 到 b8b7 到 b4b3 到 b0</div><table><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table><div>年月</div></div> 例子: 七月, 1993 9307H																		D9025																
SD211	时钟数据	时钟数据 (日期, 小时)	● 日期和小时如下所示以 BCD 码存储在 SD211 中: <div><div>b15 到 b12b11 到 b8b7 到 b4b3 到 b0</div><table><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table><div>日期小时</div></div> 例子: 31号, 10上午 3110H																	S/U(请求)	D9026	○ +Rem															
SD212	时钟数据	时钟数据 (分钟, 秒)	● 分钟和秒(小时之后)如下所示以 BCD 码存储在 SD212: <div><div>b15 到 b12b11 到 b8b7 到 b4b3 到 b0</div><table><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table><div>分钟秒</div></div> 例子: 35分, 48秒 3548H																		D9027																
SD213	时钟数据	时钟数据 (星期几)	● 以如下所示的格式以 BCD 码存储年(两位数)和星期到 SD213。 <div><div>b15 到 b12b11 到 b8b7 到 b4b3 到 b0</div><table><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table><div>↑↑</div><div>年的高位数 (0到99)</div><table><tr><th>星期几</th></tr><tr><td>0</td><td>星期天</td></tr><tr><td>1</td><td>星期一</td></tr><tr><td>2</td><td>星期二</td></tr><tr><td>3</td><td>星期三</td></tr><tr><td>4</td><td>星期四</td></tr><tr><td>5</td><td>星期五</td></tr><tr><td>6</td><td>星期六</td></tr></table></div> 例子: 1993, 星期五 1905H																	星期几	0	星期天	1	星期一	2	星期二	3	星期三	4	星期四	5	星期五	6	星期六	S/U(请求)	D9028	○ +Rem
星期几																																					
0	星期天																																				
1	星期一																																				
2	星期二																																				
3	星期三																																				
4	星期四																																				
5	星期五																																				
6	星期六																																				

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定 定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU	
SD220	LED 显示数据	显示指示器数据	● LED 显示的 ASCII 数据(16 字符)存储在这里。	S(当被改 变时)	新增	○	
SD221			b15 到 b8 b7 到 b0				
SD222			SD220 从右边起第15个字符 从右边起第16个字符				
SD223			SD221 从右边起第13个字符 从右边起第14个字符				
SD224			SD222 从右边起第11个字符 从右边起第12个字符				
SD225			SD223 从右边起第9个字符 从右边起第10个字符				
SD226			SD224 从右边起第7个字符 从右边起第8个字符				
SD227			SD225 从右边起第5个字符 从右边起第6个字符				
SD228			SD226 从右边起第3个字符 从右边起第4个字符				
SD229	SD227 从右边起第1个字符 从右边起第2个字符						
SD235	被执行在线模块改变的单元	被执行在线模块改变的单元的头 I/O 号码+10H	● 10H 被加到, 被执行在线模块改变的单元的头 I/O 号码的值上。	S(处于在线模块交换过程中)	新增	○	
SD240	基本模式	0:自动模式 1:详细模式	● 存储此基本模式。	S(初始化)	新增	○+Rem	
SD241	扩展基板的数量	0:只有主基板 1 到 7:扩展基板的数量	● 存储被安装的扩展基板的最大数量。	S(初始化)	新增		
SD242	A/Q 基板区别	基板类型区别 0:QA * * B 被安装 (A 模式) 1:Q * * B 被安装 (Q 模式)	<div><div><div>b7 到 b2 b1 b0</div><div>固定为0 到 到 到 到</div><div><div>主基板</div><div>第1扩展基板</div><div>第2扩展基板</div><div>到</div><div>第7扩展基板</div></div></div><div>当没有安装基板时固定为0</div></div> <td>S(初始化)</td> <td>新增</td>	S(初始化)	新增		
SD243	基板插槽的数量	基板插槽的数量	<div><div>b15 到 b12 b11 到 b8 b7 到 b4 b3 到 b0</div><div>SD243 扩展3 扩展2 扩展1 主</div><div>SD244 扩展7 扩展6 扩展5 扩展4</div></div> <div>● 如上所示, 每个区域存储了安装的插槽数量</div>	S(初始化)	新增	○+Rem	
SD244							
SD250	安装的最大 I/O	安装的最大 I/O 号码	● 当 SM250 从 OFF 变为 ON, 安装模块的最后一个 I/O 号码的头 2 位数字加 1, 以 BIN 码存储。	S(请求 END)	新增	○+Rem	
SD254	MELSECNET/10 (H) 信息	安装模块的数量	● 表示在 MELSECNET/H 上安装的模块的数量	S(初始化)	新增	○	
SD255		来自第 1 块模块的信息	I/O 号码				● 安装的 MELSECNET/H 模块的 I/O 号码
SD256			网络号码				● 安装的 MELSECNET/H 模块的网络号码
SD257			组号码				● 安装的 MELSECNET/H 模块的组号码
SD258			站点号码				● 安装的 MELSECNET/H 模块的站点号码
SD259			备用信息				● 在备用站点中, 备用站点的模块号码被存储。(1 到 4)
SD260 到 SD264		来自第 2 块模块的信息	● 组态和第 2 块模块的组态一样				
SD265 到 SD269		来自第 3 块模块的信息	● 组态和第 3 块模块的组态一样。				
SD270 到 SD274		来自第 4 块模块的信息	● 组态和第 4 块模块的组态一样。				

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定 时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD280	CC-Link 错误	错误检测状态	<ul style="list-style-type: none">● 当安装的 CC-Link 模块的 Xn0 变为 1 (ON)，相应的站点的位变为 1 (ON)。● 当安装的 CC-Link 模块的 Xn1 或 XnF 中的任何一个变为 OFF，相应的站点的位变为 1 (ON)。● 变为 1 (ON)，当安装的 CC-Link 模块和 CPU 模块之间无法通信时。 	S (错误)	新增	QCPU 远程
SD290	设备分配 (和参数内容 相同)	为 X 分配的点数	● 存储当前为 X 设备设置的点数	S (初始化)	新增	○ + Rem
SD291		为 Y 分配的点数	● 存储当前为 Y 设备设置的点数			
SD292		为 M 分配的点数	● 存储当前为 M 设备设置的点数			
SD293		为 L 分配的点数	● 存储当前为 L 设备设置的点数			
SD294		为 B 分配的点数	● 存储当前为 B 设备设置的点数			○ + Rem
SD295		为 F 分配的点数	● 存储当前为 F 设备设置的点数			○
SD296		为 SB 分配的点数	● 存储当前为 SB 设备设置的点数			○ + Rem
SD297		为 V 分配的点数	● 存储当前为 V 设备设置的点数			○
SD298		为 S 分配的点数	● 存储当前为 S 设备设置的点数			○ + Rem
SD299		为 T 分配的点数	● 存储当前为 T 设备设置的点数			○
SD300		为 ST 分配的点数	● 存储当前为 ST 设备设置的点数			○
SD301		为 C 分配的点数	● 存储当前为 C 设备设置的点数			○
SD302		为 D 分配的点数	● 存储当前为 D 设备设置的点数			○
SD303		为 W 分配的点数	● 存储当前为 W 设备设置的点数			○
SD304		为 SW 分配的点数	● 存储当前为 SW 设备设置的点数	S (初始化)	新增	○ + Rem
SD315	为通信处理 保留的时间	为通信处理保留的 时间	保留和 GX Developer 或其它设备进行通信处理的指定时间。 指定的值越大，和其它设备 (GX Developer，串行通信单元) 通信的响应时间就变得越短。 扫描时间增加指定的时间。设置范围：1 到 100ms 如果指定的值超出上面的范围，则被假定为没有设置。	END 处理	新增	○

特殊寄存器列表(续表)

号码	名称	含义		解释	设定者 (当被设定 定时)	相应的 ACPU D9 <div><div></div><div></div><div></div></div>	相应的 CPU
SD340	以太网信息	安装的模块数量		● 表示安装的以太网模块的数量	S(初始化)	新增	○+Rem
SD341		第 1 个模块的信息	I/O 号码	● 安装的以太网模块的 I/O 号码			
SD342			网络号码	● 安装的以太网模块的网络号码			
SD343			组号码	● 安装的以太网模块的组号码			
SD344			站号码	● 安装的以太网模块的站号码			
SD345 到 SD346			空	● 空 (第 1 个模块的以太网 IP 地址存储在缓冲存储区内。)			
SD347		空	● 空 (第 1 个模块的以太网错误代码由错误 RD 指令读取。)				
SD348 到 SD354		第 2 个模块的信息		● 组态和第 1 个模块的组态相同。	S(初始化)	新增	
SD355 到 SD361		第 3 个模块的信息		● 组态和第 1 个模块的组态相同。			
SD362 到 SD368		以太网信息	第 4 个模块的信息		● 组态和第 1 个模块的组态相同。		
SD380	以太网指令接收状态	第 1 个模块的指令接收状态	<div><div><div>b15</div><div>b8</div><div>b7</div><div>b6</div><div>b5</div><div>b4</div><div>b3</div><div>b2</div><div>b1</div><div>b0</div></div><div><div>0 到 0</div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div><div>未使用</div><div>通道1的指令接收状态</div><div>通道2的指令接收状态</div><div>通道3的指令接收状态</div><div>通道4的指令接收状态</div><div>通道5的指令接收状态</div><div>通道6的指令接收状态</div><div>通道7的指令接收状态</div><div>通道8的指令接收状态</div><div>ON: 已接收 (通道被使用)</div><div>OFF: 未接收 (通道未被使用)</div></div>		S(初始化)	新增	QnA
SD381	以太网指令接收状态	来自第 2 个模块的信息	● 组态和第 1 个模块的组态相同。	S(初始化)	新增		
SD382		来自第 3 个模块的信息	● 组态和第 1 个模块的组态相同。				
SD383		来自第 4 个模块的信息	● 组态和第 1 个模块的组态相同。				
SD392	软件版本	内部系统软件版本		<div>● 以 ASCII 码存储内部系统软件版本。</div> <div><div><div>高字节</div><div>低字节</div></div><div>低字节位置的数据是不确定的。 软件版本存储在高字节位置。</div><div>对于版本“A”，举例来说，“41H”被存储。</div><div>注意：内部系统软件版本可能和打印在包装上的版本符号表示的版本不一样。</div></div>	S(初始化)	D9060	
SD395	多 PLC 号码	多 PLC 号码		● 在一个多 PLC 系统的组态中，主机 CPU 的 PLC 号码被存储。 1 号 PLC: 1, 2 号 PLC: 2, 3 号 PLC: 3, 4 号 PLC: 4	S(初始化)	新增	○

(3) 系统时钟 / 计数器

号码	名称	含义	解释	设定者 (当被设定)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD412	1 秒计数器	1-秒单位内的计数次数	● 在可编程控制器 CPU 模块变为 RUN 之后，每秒钟加 1。 ● 计数在从 0 到 32767 到-32768 再到 0 之间重复。	S (状态改变)	D9022	○
SD414	2n 秒时钟设置	2n 秒时钟单元	● 存储 2n 秒时钟的值 n(默认为 30)。 ● 设置可以在 1 到 32767 之间进行。	U	新增	
SD415	2nms 时钟设置	2nms 时钟单元	● 存储 2nms 时钟的值 n(默认为 30)。 ● 设置可以在 1 到 32767 之间进行。	U	新增	
SD420	扫描计数器	每个扫描周期内计数次数	● 在 CPU 模块变为 RUN 之后，在扫描执行类型程序的每个扫描周期此计数器增加 1。* ● 计数在从 0 到 32767 到-32768 再到 0 之间重复。	S(每个 END 处理)	新增	
SD430	低速扫描计数器	每个扫描周期内计数次数	● 在 CPU 模块变为 RUN 之后，在低速执行类型程序的每个扫描周期此计数器增加 1。 ● 计数在从 0 到 32767 到-32768 再到 0 之间重复。 ● 只用于低速执行类型程序。	S(每个 END 处理)	新增	

*：在初始化执行类型程序中不被计数

特殊寄存器列表

(4) 扫描信息

号码	名称	含义	解释	设定者 (当被设定 时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD500	执行程序号码	处于执行中的程序号码	● 当前正被执行的程序的程序号码被以 BIN 值存储。	S (状态改变)	新增	○
SD510	低速执行类型 程序号码	处于执行中的低速执行 类型程序号码	● 当前正被执行的低速执行类型程序的程序号码被以 BIN 值 存储。 ● 只有在 SM510 为 ON 时才使能。	S(每个 END 处理)	新增	
SD520	当前扫描时间	当前扫描时间 (以 1ms 为单位)	● 存储当前扫描时间(以 1ms 为单位) 范围: 0 到 65535	S(每个 END 处理)	D9017 格式改变	
SD521		当前扫描时间 (以 100μs 为单位)	● 存储当前扫描时间 (以 100μs 为单位) 范围: 00000 到 900 (例子) 当前扫描时间 23.6 ms, 将按如下所示的方式存储: D520=23 D521=600	S(每个 END 处理)	新增	
SD522	初始扫描时间	初始扫描时间 (以 1ms 为单位)	● 存储初始执行类型程序的扫描时间到 SD522 和 SD523 中 (测量是以 100μs 为单位进行的) SD522: 存储此 ms 位置(存储范围: 0 到 65535) SD523: 存储此 μs 位置(存储范围: 0 到 900)	S(第一个 END 处理)	新增	
SD523		初始扫描时间 (以 100μs 为单位)				
SD524	最小扫描时间	最小扫描时间 (以 1ms 为单位)	● 存储除初始执行类型程序外, 其它类型程序的扫描时间的 最小值到 SD524 和 SD525。(测量是 100μs 为单位进行的) SD524: 存储此 ms 位置(存储范围: 0 到 65535) SD525: 存储此 μs 位置(存储范围: 0 到 900)	S(每个 END 处理)	D9018 格式改变	
SD525		最小扫描时间 (以 100μs 为单位)		S(每个 END 处理)	新增	
SD526	最大扫描时间	最大扫描时间 (以 1ms 为单位)	● 存储除初始执行类型程序外, 其它类型程序的扫描时间的 最大值到 SD526 和 SD527。(测量是 100μs 为单位进行的) SD526: 存储此 ms 位置(存储范围: 0 到 65535) SD527: 存储此 μs 位置(存储范围: 0 到 900)	S(每个 END 处理)	D9019 格式改变	
SD527		最大扫描时间 (以 100μs 为单位)			新增	
SD528	当前扫描时间 于低速执行类型 程序	当前扫描时间 (以 1ms 为单位)	● 存储低速执行类型程序的当前扫描时间到 SD528 和 SD529 (测量是以 100μs 为单位进行的) SD528: 存储此 ms 位置(存储范围: 0 到 65535) SD529: 存储此 μs 位置(存储范围: 0 到 900)	S(每个 END 处理)	新增	
SD529		当前扫描时间 (以 100μs 为单位)				
SD532	最小扫描时间 用于低速执行 类型程序	最小扫描时间 (以 1ms 为单位)	● 存储此低速执行类型程序的扫描时间的最小值到 SD532 和 SD533(测量是以 100μs 为单位进行的) SD532: 存储此 ms 位置(存储范围: 0 到 65535) SD533: 存储此 μs 位置(存储范围: 0 到 900)	S(每个 END 处理)	新增	
SD533		最小扫描时间 (以 100μs 为单位)				
SD534	最大扫描时间 用于低速执行 类型程序	最大扫描时间 (以 1ms 为单位)	● 存储此低速执行类型程序除第一个扫描周期外, 其它扫描 时间的最大值到 SD534 和 SD535。(测量是以 100μs 为单 位进行的) SD534: 存储此 ms 位置(存储范围: 0 到 65535) SD535: 存储此 μs 位置(存储范围: 0 到 900)	S(每个 END 处理)	新增	
SD535		最大扫描时间 (以 100μs 为单位)				
SD540	END 处理时间	END 处理时间 (以 1ms 为单位)	● 存储扫描执行类型程序一个扫描结束到下一个扫描开始 的时间到 SD540 和 SD541。(测量是以 100μs 为单位进行的) SD540: 存储此 ms 位置(存储范围: 0 到 65535) SD541: 存储此 μs 位置(存储范围: 0 到 900)	S(每个 END 处理)	新增	
SD541		END 处理时间 (以 100μs 为单位)				
SD542	固定扫描等待	固定扫描等待时间 (以 1ms 为单位)	● 存储此固定扫描设置的等待时间到 SD542 中 SD543(测量是 以 100μs 为单位进行的) SD542: 存储此 ms 位置(存储范围: 0 到 65535) SD543: 存储此 μs 位置(存储范围: 0 到 900)	S(第一个 END 处理)	新增	
SD543	时间	固定扫描等待时间 (以 100μs 为单位)				

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定 时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD544	低速执行类型 程序的累计执 行时间	低速执行类型程序的 累计执行时间 (以 1ms 为单位)	● 存储此低速执行类型程序的累计执行时间到 SD544 和 SD545(测量是以 100μs 为单位进行的) SD544: 存储此 ms 位置(存储范围: 0 到 65535) SD545: 存储此 ms 位置(存储范围: 0 到 900) ● 在一个低速扫描结束后清为 0。	S(每个 END 处理)	新增	○
SD545		低速执行类型程序的 累计执行时间 (以 100μs 为单位)				
SD546	低速执行类型 程序的执行时 间	低速执行类型程序的 执行时间 (以 1ms 为单位)	● 存储低速执行类型程序一个扫描周期内的执行时间到 SD546 和 SD547(测量是以 100μs 为单位进行的) SD546: 存储此 ms 位置(存储范围: 0 到 65535) SD547: 存储此 μs 位置(存储范围: 0 到 900) ● 每个扫描周期都存储。	S(每个 END 处理)	新增	
SD547		低速执行类型程序的 执行时间 (以 100μs 为单位)				
SD548	扫描程序执行 时间	扫描程序执行时间 (以 1ms 为单位)	● 存储扫描执行类型程序一个扫描周期内的执行时间到 SD548 和 SD549。(测量是以 100ms 为单位进行的) SD548: 存储此 ms 位置(存储范围: 0 到 65535) SD549: 存储此 μs 位置(存储范围: 0 到 900) ● 每个扫描周期都存储。	S(每个 END 处理)	新增	
SD549		扫描程序执行时间 (以 100μs 为单位)				
SD550	服务间隔测量 模块	单元/模块号码	● 设置用于测量服务间隔的模块的 I/O 号码。	U	新增	○+Rem
SD551	服务间隔时间	模块服务间隔 (以 1ms 为单位)	● 当 SM551 变为 ON 时,存储 SD550 中指定模块的服务间隔到 SD551 和 SD552。(测量是以 100ms 为单位进行的) SD551: 存储此 ms 位置(存储范围: 0 到 65535) SD552: 存储此 μs 位置(存储范围: 0 到 900)	S(请求)	新增	
SD552		模块服务间隔 (以 100μs 为单位)				

(5) 存储卡

特殊寄存器列表

号码	名称	含义	解释	设定者 (当被设定 时)	相应的 ACPU D9 <div><div></div><div></div><div></div></div>	相应的 CPU
SD600	存储卡模块	存储卡模块	<div><div><div>表示安装的存储卡 A 模块</div><div><div><div>b15 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div><div>0</div></div></div><div><div><div>驱动器1 (RAM) 模块</div><div>0:不存在 1:SRAM卡</div></div><div><div>驱动器2 (ROM) 模块</div><div>0:不存在 2:ATA卡 3:闪存卡</div></div></div></div></div> <td>S(初始化和 卡件拆除)</td> <td>新增</td> <td>QCPU</td>	S(初始化和 卡件拆除)	新增	QCPU
SD602	驱动器 1 (标准 RAM) 容量	驱动器 1 容量	<div><div><div>驱动器 1 容量以 1k 字节为单元存储。</div></div></div>	S(初始化和 卡件拆除)	新增	QCPU
SD603	驱动器 2 (标准 ROM) 容量	驱动器 2 容量	<div><div><div>驱动器 2 容量以 1k 字节为单元存储。</div></div></div>	S(初始化和 卡件拆除)	新增	QCPU
SD604	存储卡使用 情况	存储卡使用 情况	<div><div><div><div><div><div>存储卡的使用情况以位模式存储(当为 ON 时处于使用中)</div><div>这些位模式的含义如下所示:</div></div><div><div><div>b0: 导入操作(QBT)</div><div>b1: 设备注释(QCD)</div><div>b2: 设备注释(QCD)</div><div>b3: 设备初始值(QDI)</div><div>b4: 文件寄存器 R(QDR)</div><div>b5: 跟踪(QTS)</div><div>b6: 未使用</div><div>b7: 未使用</div></div><div><div><div>b8: 未使用</div><div>b9: CPU 故障历史(QFD)</div><div>b10: SFC 程序(QTS)</div><div>b11: 本地设备(QDL)</div><div>b12: 未使用</div><div>b13: 未使用</div><div>b14: 未使用</div><div>b15: 未使用</div></div></div></div></div></div></div><td>S (状态改变)</td><td>新增</td><td>QCPU</td></div>	S (状态改变)	新增	QCPU
SD620	驱动器 3/4 模块	驱动器 3/4 模块	<div><div><div><div>表示安装的驱动器 3/4 模块</div><div><div><div>b15 到 b8</div><div>b7 到 b4</div><div>b3 到 b0</div></div><div><div>0</div></div></div><div><div><div>驱动器3 (标准RAM)</div><div>固定为“0”</div></div><div><div>驱动器4 (标准ROM)</div><div>固定为“3”</div></div></div></div></div></div> <td>S(初始化)</td> <td>新增</td> <td>QCPU</td>	S(初始化)	新增	QCPU
SD622	驱动器 3 (RAM) 容量	驱动器 3 容量	<div><div><div>驱动器 3 容量以 1k 字节为单元存储。</div></div></div>	S(初始化)	新增	QCPU
SD623	驱动器 4 (ROM) 容量	驱动器 4 容量	<div><div><div>驱动器 4 容量以 1k 字节为单元存储。</div></div></div>	S(初始化)	新增	QCPU
SD624	驱动器 3/4 使用 情况	驱动器 3/4 使用情况	<div><div><div><div><div><div>驱动器 3/4 的使用情况以位模式存储。(当为 ON 时处于使用中)</div><div>这些位模式的含义如下所示:</div></div><div><div><div>b0: 导入操作(QBT)</div><div>b1: 参数(QPA)</div><div>b2: 设备注释(QCD)</div><div>b3: 设备初始值(QDI)</div><div>b4: 文件寄存器 R(QDR)</div><div>b5: 跟踪(QTS)</div><div>b6: 未使用</div><div>b7: 未使用</div></div><div><div><div>b8: 未使用</div><div>b9: CPU 故障历史(QFD)</div><div>b10: SFC 跟踪(QTR)</div><div>b11: 本地设备(QDL)</div><div>b12: 未使用</div><div>b13: 未使用</div><div>b14: 未使用</div><div>b15: 未使用</div></div></div></div></div></div></div><td>S (状态改变)</td><td>新增</td><td>QCPU</td></div>	S (状态改变)	新增	QCPU

特殊寄存器列表(续表)

号码	名称	含义	解释	设定者 (当被设定 定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU																																										
SD640	文件寄存器驱动器	驱动器号码:	● 存储被文件寄存器使用的驱动器号码	S(初始化)	新增	○																																										
SD641	文件寄存器文件名称	文件寄存器文件名称	● 以 ASCII 码存储在参数上选择的或者使用 QDRSET 指令的文件寄存器文件名称(带扩展)。 <table><tr><td>b15</td><td>到</td><td>b8</td><td>b7</td><td>到</td><td>b0</td></tr><tr><td>SD641</td><td>第二个字符</td><td></td><td></td><td>第一个字符</td><td></td></tr><tr><td>SD642</td><td>第四个字符</td><td></td><td></td><td>低三个字符</td><td></td></tr><tr><td>SD643</td><td>第六个字符</td><td></td><td></td><td>第五个字符</td><td></td></tr><tr><td>SD644</td><td>第八个字符</td><td></td><td></td><td>第七个字符</td><td></td></tr><tr><td>SD645</td><td>扩展的第一个字符</td><td></td><td></td><td>2E#(.)</td><td></td></tr><tr><td>SD646</td><td>扩展的第三个字符</td><td></td><td></td><td>扩展的第二个字符</td><td></td></tr></table>	b15	到		b8	b7	到	b0	SD641	第二个字符			第一个字符		SD642	第四个字符			低三个字符		SD643	第六个字符			第五个字符		SD644	第八个字符			第七个字符		SD645	扩展的第一个字符			2E#(.)		SD646	扩展的第三个字符			扩展的第二个字符		S(初始化)	新增
b15				到	b8		b7	到	b0																																							
SD641				第二个字符				第一个字符																																								
SD642				第四个字符				低三个字符																																								
SD643				第六个字符				第五个字符																																								
SD644				第八个字符				第七个字符																																								
SD645	扩展的第一个字符			2E#(.)																																												
SD646	扩展的第三个字符			扩展的第二个字符																																												
SD642																																																
SD643																																																
SD644																																																
SD645																																																
SD646																																																
SD647	文件寄存器容量	文件寄存器容量	● 以 1k 字为单元存储当前选择的文件寄存器的数据容量。	S(状态 改变)	新增																																											
SD648	文件寄存器块 号码	文件寄存器块 号码	● 存储当前选择的文件寄存器块号码。	S(状态 改变)	D9035																																											
SD650	注释驱动器	注释驱动器号 码	● 存储在参数上或者被 QCDSET 指令选择的注释驱动器号码。	S(状态 改变)	新增																																											
SD651	注释文件名称	注释文件名称	● 以 ASCII 码存储在参数上或者被 QCDSET 指令选择的注释文件名称 (带扩展) <table><tr><td>b15</td><td>到</td><td>b8</td><td>b7</td><td>到</td><td>b0</td></tr><tr><td>SD651</td><td>第二个字符</td><td></td><td></td><td>第一个字符</td><td></td></tr><tr><td>SD652</td><td>第四个字符</td><td></td><td></td><td>第三个字符</td><td></td></tr><tr><td>SD653</td><td>第六个字符</td><td></td><td></td><td>第五个字符</td><td></td></tr><tr><td>SD654</td><td>第八个字符</td><td></td><td></td><td>第七个字符</td><td></td></tr><tr><td>SD655</td><td>扩展的第一个字符</td><td></td><td></td><td>2E#(.)</td><td></td></tr><tr><td>SD656</td><td>扩展的第三个字符</td><td></td><td></td><td>扩展的第二个字符</td><td></td></tr></table>	b15	到	b8	b7	到	b0	SD651	第二个字符			第一个字符		SD652	第四个字符			第三个字符		SD653	第六个字符			第五个字符		SD654	第八个字符			第七个字符		SD655	扩展的第一个字符			2E#(.)		SD656	扩展的第三个字符			扩展的第二个字符		S(状态 改变)	新增	
b15				到	b8	b7	到	b0																																								
SD651				第二个字符			第一个字符																																									
SD652				第四个字符			第三个字符																																									
SD653				第六个字符			第五个字符																																									
SD654				第八个字符			第七个字符																																									
SD655	扩展的第一个字符			2E#(.)																																												
SD656	扩展的第三个字符			扩展的第二个字符																																												
SD652																																																
SD653																																																
SD654																																																
SD655																																																
SD656																																																
SD660	导入操作指定 文件	导入指定文件 驱动器号码	● 存储导入指定文件(*.QBT)存储处的驱动器号码。	S(初始化)	新增	○																																										
SD661		导入指定文件 的文件名称	● 存储导入指定文件(*.QBT) 的文件名称。 <table><tr><td>b15</td><td>到</td><td>b8</td><td>b7</td><td>到</td><td>b0</td></tr><tr><td>SD661</td><td>第二个字符</td><td></td><td></td><td>第一个字符</td><td></td></tr><tr><td>SD662</td><td>第四个字符</td><td></td><td></td><td>第三个字符</td><td></td></tr><tr><td>SD663</td><td>第六个字符</td><td></td><td></td><td>第五个字符</td><td></td></tr><tr><td>SD664</td><td>第八个字符</td><td></td><td></td><td>第七个字符</td><td></td></tr><tr><td>SD665</td><td>扩展的第一个字符</td><td></td><td></td><td>2E#(.)</td><td></td></tr><tr><td>SD666</td><td>扩展的第三个字符</td><td></td><td></td><td>扩展的第二个字符</td><td></td></tr></table>	b15	到		b8	b7	到	b0	SD661	第二个字符			第一个字符		SD662	第四个字符			第三个字符		SD663	第六个字符			第五个字符		SD664	第八个字符			第七个字符		SD665	扩展的第一个字符			2E#(.)		SD666	扩展的第三个字符			扩展的第二个字符		S(初始化)	新增
b15				到	b8		b7	到	b0																																							
SD661				第二个字符				第一个字符																																								
SD662				第四个字符				第三个字符																																								
SD663				第六个字符				第五个字符																																								
SD664				第八个字符				第七个字符																																								
SD665	扩展的第一个字符			2E#(.)																																												
SD666	扩展的第三个字符			扩展的第二个字符																																												
SD662																																																
SD663																																																
SD664																																																
SD665																																																
SD666																																																

(6) 和寄存器相关的指令

号码	名称	含义	解释	设定者 (当被设定 时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU																																																																																																																																																																																																																																							
SD705	掩码模式	掩码模式	● 在块操作的过程中，将 SM705 变为 ON，就可以使用存储在 SD705 上(或者在 SD705 和 SD706 上，如果使用的是双字)的掩码模式，用掩码值来操作数据块内的所有数据了。	U	新增	○																																																																																																																																																																																																																																							
SD706																																																																																																																																																																																																																																													
SD715	IMASK 指令掩 码模式	掩码模式	● 使用 IMASK 指令掩码的模式，按照如下所示的样式存储： <table><tr><td></td><td>b15</td><td>到</td><td>b1</td><td>b0</td></tr><tr><td>SD715</td><td>115</td><td>到</td><td>11</td><td>10</td></tr><tr><td>SD716</td><td>131</td><td>到</td><td>117</td><td>116</td></tr><tr><td>SD717</td><td>147</td><td>到</td><td>133</td><td>132</td></tr></table>		b15	到	b1	b0	SD715	115	到	11	10	SD716	131	到	117	116	SD717	147	到	133	132	S(在执行 过程中)	新增	○																																																																																																																																																																																																																			
				b15	到	b1	b0																																																																																																																																																																																																																																						
SD715				115	到	11	10																																																																																																																																																																																																																																						
SD716				131	到	117	116																																																																																																																																																																																																																																						
SD717	147	到	133	132																																																																																																																																																																																																																																									
SD716																																																																																																																																																																																																																																													
SD717																																																																																																																																																																																																																																													
SD718	报警器	报警器	● 在 A 一系列程序中用作报警器的代替者。	S/U	新增																																																																																																																																																																																																																																								
SD719																																																																																																																																																																																																																																													
SD720	用于 PLOAD 指 令的程序号码 指定	用于 PLOAD 指 令的程序号码 指定	● 在被指定时，存储被 PLOAD 指令装载的程序的程序号码。 ● 指定范围：1 到 124	U	新增	○																																																																																																																																																																																																																																							
SD736	PKEY 输入	PKEY 输入	● 临时存储通过使用 PKEY 指令输入键盘数据的 SD。	S(在执行 过程中)	新增	○																																																																																																																																																																																																																																							
SD738	消息存储	消息存储	● 存储由 MSG 指令指定的消息。 <table><tr><td></td><td>b15</td><td>到</td><td>b8</td><td>b7</td><td>到</td><td>b0</td></tr><tr><td>SD738</td><td>第2个字符</td><td></td><td></td><td></td><td></td><td>第1个字符</td></tr><tr><td>SD739</td><td>第4个字符</td><td></td><td></td><td></td><td></td><td>第3个字符</td></tr><tr><td>SD740</td><td>第6个字符</td><td></td><td></td><td></td><td></td><td>第5个字符</td></tr><tr><td>SD741</td><td>第8个字符</td><td></td><td></td><td></td><td></td><td>第7个字符</td></tr><tr><td>SD742</td><td>第10个字符</td><td></td><td></td><td></td><td></td><td>第9个字符</td></tr><tr><td>SD743</td><td>第12个字符</td><td></td><td></td><td></td><td></td><td>第11个字符</td></tr><tr><td>SD744</td><td>第14个字符</td><td></td><td></td><td></td><td></td><td>第13个字符</td></tr><tr><td>SD745</td><td>第16个字符</td><td></td><td></td><td></td><td></td><td>第15个字符</td></tr><tr><td>SD746</td><td>第18个字符</td><td></td><td></td><td></td><td></td><td>第17个字符</td></tr><tr><td>SD747</td><td>第20个字符</td><td></td><td></td><td></td><td></td><td>第19个字符</td></tr><tr><td>SD748</td><td>第22个字符</td><td></td><td></td><td></td><td></td><td>第21个字符</td></tr><tr><td>SD749</td><td>第24个字符</td><td></td><td></td><td></td><td></td><td>第23个字符</td></tr><tr><td>SD750</td><td>第26个字符</td><td></td><td></td><td></td><td></td><td>第25个字符</td></tr><tr><td>SD751</td><td>第28个字符</td><td></td><td></td><td></td><td></td><td>第27个字符</td></tr><tr><td>SD752</td><td>第30个字符</td><td></td><td></td><td></td><td></td><td>第29个字符</td></tr><tr><td>SD753</td><td>第32个字符</td><td></td><td></td><td></td><td></td><td>第31个字符</td></tr><tr><td>SD754</td><td>第34个字符</td><td></td><td></td><td></td><td></td><td>第33个字符</td></tr><tr><td>SD755</td><td>第36个字符</td><td></td><td></td><td></td><td></td><td>第35个字符</td></tr><tr><td>SD756</td><td>第38个字符</td><td></td><td></td><td></td><td></td><td>第37个字符</td></tr><tr><td>SD757</td><td>第40个字符</td><td></td><td></td><td></td><td></td><td>第39个字符</td></tr><tr><td>SD758</td><td>第42个字符</td><td></td><td></td><td></td><td></td><td>第41个字符</td></tr><tr><td>SD759</td><td>第44个字符</td><td></td><td></td><td></td><td></td><td>第43个字符</td></tr><tr><td>SD760</td><td>第46个字符</td><td></td><td></td><td></td><td></td><td>第45个字符</td></tr><tr><td>SD761</td><td>第48个字符</td><td></td><td></td><td></td><td></td><td>第47个字符</td></tr><tr><td>SD762</td><td>第50个字符</td><td></td><td></td><td></td><td></td><td>第49个字符</td></tr><tr><td>SD763</td><td>第52个字符</td><td></td><td></td><td></td><td></td><td>第51个字符</td></tr><tr><td>SD764</td><td>第54个字符</td><td></td><td></td><td></td><td></td><td>第53个字符</td></tr><tr><td>SD765</td><td>第56个字符</td><td></td><td></td><td></td><td></td><td>第55个字符</td></tr><tr><td>SD766</td><td>第58个字符</td><td></td><td></td><td></td><td></td><td>第57个字符</td></tr><tr><td>SD767</td><td>第60个字符</td><td></td><td></td><td></td><td></td><td>第59个字符</td></tr><tr><td>SD768</td><td>第62个字符</td><td></td><td></td><td></td><td></td><td>第61个字符</td></tr><tr><td>SD769</td><td>第64个字符</td><td></td><td></td><td></td><td></td><td>第63个字符</td></tr></table>		b15	到	b8	b7	到	b0	SD738	第2个字符					第1个字符	SD739	第4个字符					第3个字符	SD740	第6个字符					第5个字符	SD741	第8个字符					第7个字符	SD742	第10个字符					第9个字符	SD743	第12个字符					第11个字符	SD744	第14个字符					第13个字符	SD745	第16个字符					第15个字符	SD746	第18个字符					第17个字符	SD747	第20个字符					第19个字符	SD748	第22个字符					第21个字符	SD749	第24个字符					第23个字符	SD750	第26个字符					第25个字符	SD751	第28个字符					第27个字符	SD752	第30个字符					第29个字符	SD753	第32个字符					第31个字符	SD754	第34个字符					第33个字符	SD755	第36个字符					第35个字符	SD756	第38个字符					第37个字符	SD757	第40个字符					第39个字符	SD758	第42个字符					第41个字符	SD759	第44个字符					第43个字符	SD760	第46个字符					第45个字符	SD761	第48个字符					第47个字符	SD762	第50个字符					第49个字符	SD763	第52个字符					第51个字符	SD764	第54个字符					第53个字符	SD765	第56个字符					第55个字符	SD766	第58个字符					第57个字符	SD767	第60个字符					第59个字符	SD768	第62个字符					第61个字符	SD769	第64个字符					第63个字符	S(在执行 过程中)	新增	○
				b15	到	b8	b7	到	b0																																																																																																																																																																																																																																				
SD738				第2个字符					第1个字符																																																																																																																																																																																																																																				
SD739				第4个字符					第3个字符																																																																																																																																																																																																																																				
SD740				第6个字符					第5个字符																																																																																																																																																																																																																																				
SD741				第8个字符					第7个字符																																																																																																																																																																																																																																				
SD742				第10个字符					第9个字符																																																																																																																																																																																																																																				
SD743				第12个字符					第11个字符																																																																																																																																																																																																																																				
SD744				第14个字符					第13个字符																																																																																																																																																																																																																																				
SD745				第16个字符					第15个字符																																																																																																																																																																																																																																				
SD746				第18个字符					第17个字符																																																																																																																																																																																																																																				
SD747				第20个字符					第19个字符																																																																																																																																																																																																																																				
SD748				第22个字符					第21个字符																																																																																																																																																																																																																																				
SD749				第24个字符					第23个字符																																																																																																																																																																																																																																				
SD750				第26个字符					第25个字符																																																																																																																																																																																																																																				
SD751				第28个字符					第27个字符																																																																																																																																																																																																																																				
SD752				第30个字符					第29个字符																																																																																																																																																																																																																																				
SD753				第32个字符					第31个字符																																																																																																																																																																																																																																				
SD754				第34个字符					第33个字符																																																																																																																																																																																																																																				
SD755				第36个字符					第35个字符																																																																																																																																																																																																																																				
SD756				第38个字符					第37个字符																																																																																																																																																																																																																																				
SD757				第40个字符					第39个字符																																																																																																																																																																																																																																				
SD758				第42个字符					第41个字符																																																																																																																																																																																																																																				
SD759				第44个字符					第43个字符																																																																																																																																																																																																																																				
SD760				第46个字符					第45个字符																																																																																																																																																																																																																																				
SD761				第48个字符					第47个字符																																																																																																																																																																																																																																				
SD762				第50个字符					第49个字符																																																																																																																																																																																																																																				
SD763				第52个字符					第51个字符																																																																																																																																																																																																																																				
SD764				第54个字符					第53个字符																																																																																																																																																																																																																																				
SD765				第56个字符					第55个字符																																																																																																																																																																																																																																				
SD766				第58个字符					第57个字符																																																																																																																																																																																																																																				
SD767				第60个字符					第59个字符																																																																																																																																																																																																																																				
SD768	第62个字符					第61个字符																																																																																																																																																																																																																																							
SD769	第64个字符					第63个字符																																																																																																																																																																																																																																							

特殊寄存器列表 (续表)

号码	名称	含义	解释	设定者 (当被设定 时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU																				
SD781 T0 SD793	IMASK 指令的 掩码模式	掩码模式	<div>● 如下所示, 存储 IMASK 指令掩码的掩码模式:</div> <div><div>b15到b1b0</div><table><tr><td>SD781</td><td>163</td><td>到</td><td>149</td><td>148</td></tr><tr><td>SD782</td><td>179</td><td>到</td><td>165</td><td>164</td></tr><tr><td></td><td colspan="4">到</td></tr><tr><td>SD793</td><td>1255</td><td>到</td><td>1241</td><td>1240</td></tr></table></div>	SD781	163	到	149	148	SD782	179	到	165	164		到				SD793	1255	到	1241	1240	S(在执行 过程中)	新增	○
SD781	163	到	149	148																						
SD782	179	到	165	164																						
	到																									
SD793	1255	到	1241	1240																						

(7) A 到 Q/QnA 转换的对应

A 系列到 Q/QnA 系列转换后，ACPU 的特殊寄存器 D9000 至 D9255 对应于特殊寄存器 SD1000 至 SD1255。

这些特殊寄存器都是由系统设置的，并且用户使用它们去设置程序数据。

需要使用这些寄存器设置数据的用户应该修改用于 Q/QnA 的特殊寄存器。

然而，在转换之前用户只能在特殊寄存器 D9200 至 D9255 上设置数据，而在转换之后用户也可以在寄存器 1200 至 1255 上设置数据。

关于 ACPU 特殊寄存器的内容的更多详细信息，参见每个 CPU 的用户手册，和 MELSECNET 和 MELSECNET/B 数据链接系统参考手册。

备注

- 关于“用于修改的特殊寄存器”栏目的补充解释
- ① 用于用于修改的特殊寄存器被指定的软元件号，修改它以对应于用于 QCPU/QnACPU 的特殊寄存器。
 - ② 用于 ☐ 被指定的软元件号，转换后的特殊寄存器能被使用。
 - ③ ☒ 被指定的软元件号没有用于 QCPU/QnACPU 的功能。

特殊寄存器列表

ACPU 特殊转换	转换后的 特殊寄存器	用于修改的 特殊寄存器	名称	含义	详细内容	相应的 CPU																																							
D9000	SD1000	—	保险丝烧坏	保险丝烧坏的模块的数量	<ul style="list-style-type: none">当保险丝烧坏的模块被探测到时，探测到的最低号码的模块的第一个 I/O 号码被存储成十六进制数据。(例子：当 Y50 到 6F 输出模块的保险丝被烧坏时，“50”被存储进十六进制)要想通过外部设备监视此号码，执行以十六进制给出的监视操作。 (当 SD1100 到 SD1107 中的所有内容都被复位到 0 时，号码被清除)对于远程 I/O 站点的输出模块也执行保险丝烧坏检查。	○																																							
D9001	SD1001	—	保险丝烧坏	保险丝烧坏的模块的数量	<ul style="list-style-type: none">当发生保险丝烧坏时，存储和设置开关号码或者基板插槽号码相对应的模块号码。 <table><tr><th colspan="2">用于 A0J2 的 I/O 模块</th><th colspan="2">扩展基板单元</th></tr><tr><th>设置开关</th><th>存储区域</th><th>基板单元插槽号码</th><th>存储区域</th></tr><tr><td>0</td><td>1</td><td>0</td><td>5</td></tr><tr><td>1</td><td>2</td><td>1</td><td>6</td></tr><tr><td>2</td><td>3</td><td>2</td><td>7</td></tr><tr><td>3</td><td>4</td><td>3</td><td>8</td></tr><tr><td>4</td><td>5</td><td></td><td></td></tr><tr><td>5</td><td>6</td><td></td><td></td></tr><tr><td>6</td><td>7</td><td></td><td></td></tr><tr><td>7</td><td>8</td><td></td><td></td></tr></table>		用于 A0J2 的 I/O 模块		扩展基板单元		设置开关	存储区域	基板单元插槽号码	存储区域	0	1	0	5	1	2	1	6	2	3	2	7	3	4	3	8	4	5			5	6			6	7			7	8	
用于 A0J2 的 I/O 模块		扩展基板单元																																											
设置开关	存储区域	基板单元插槽号码	存储区域																																										
0	1	0	5																																										
1	2	1	6																																										
2	3	2	7																																										
3	4	3	8																																										
4	5																																												
5	6																																												
6	7																																												
7	8																																												

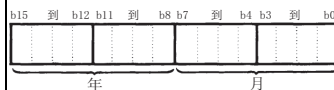
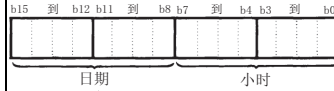
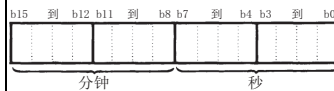
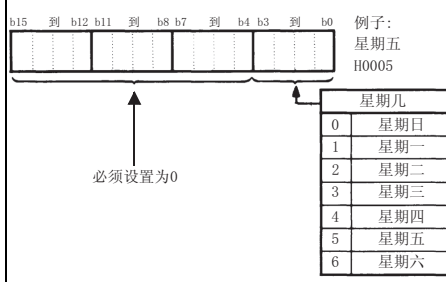
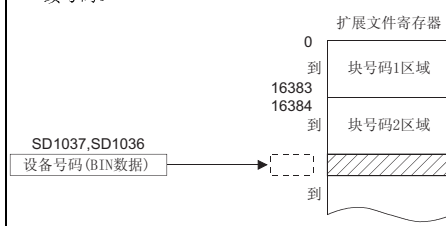

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后的 特殊寄存器	用于修改的 特殊寄存器	名称	含义	详细内容	相应的 CPU
D9002	SD1002	—	I/O 模块校验错误	I/O 模块校验错误 模块号码	<ul style="list-style-type: none"> 在电源上电时, 如果 I/O 模块, 其数据和输入的数据不同, 被探测到, 在被探测到的单元中号码最低的单元的 1/O 号码被存储成十六进制数据。(存储方法和 SD1000 的相同) 要通过外部设备监视此号码, 执行以十六进制给出的监视操作。 (当 SD1116 到 SD1123 中的所有内容都被复位到 0 时, 号码被清除。) 对于远程 I/O 终端的模块也执行保险丝烧坏检查。 	○
D9004	SD1004	—	MINI 链接错误	存储参数上的设置 状态 (模块 1 到 8)	<p> b15 到 b8 b7 到 b0 8 7 6 5 4 3 2 1 8 7 6 5 4 3 2 1 </p> <p> 和 AJ71PT32 (S3) 上的故障相对应的位变为 ON。 和 AJ71PT32 (S3) 上的信号相对应的位, 如下所示, 随着这些信号变为 ON 而变为 ON。 • 硬件错误 (X0/X20) • MINI (S3) 链接错误探测 (X6/X26) • MINI (S3) 链接通信错误 (X7/X27) </p>	QnA
D9005	SD1005	—	AC DOWN 计数器	AC DOWN 的次数	当使用 AC 供电电源模块时, 每发生 20ms 内的瞬时电源故障就增加 1。 (此值以 BIN 码存储) 当电源从 OFF 变为 ON 时此值被复位。	○
					当使用 DC 供电电源模块时, 每发生 10ms 内的瞬时电源故障就增加 1。 (此值以 BIN 码存储) 当电源从 OFF 变为 ON 时此值被复位。	QnA
D9008	SD1008	SD0	自诊断错误	自诊断错误号码	<ul style="list-style-type: none"> 当自诊断结果发现错误时, 错误号码被以 BIN 码存储。 	○
D9009	SD1009	SD62	报警器探测	发生外部故障的 F 号码	<ul style="list-style-type: none"> 如果 F0 到 255 中的一个被 OUT F 或 SET F 变为 ON, 在变为 ON 的 F 号码中最先被探测到的 F 号码, 被以 BIN 码存储。 SD62 可以由 RST F 或 LEDR 指令清除。如果有另外一个 F 号码被探测到, SD62 的清除将引起下一个号码被存储到 SD62。 	QCPU Q2AS Q2A
					<ul style="list-style-type: none"> 如果 F0 到 255 中的一个被 OUT F 或 SET F 变为 ON, 在变为 ON 的 F 号码中最先被探测到的 F 号码, 被以 BIN 码存储。 SD62 可以通过执行 RST F 或 LEDR 指令而被清除, 或者移动 CPU 模块前面的 INDICATOR RESET 开关到 ON 位置也可以清除 SD62。 如果有另外一个 F 号码被探测到, SD62 的清除将引起下一个号码被存储到 SD62。 	Q3A Q4A Q4AR
D9010	SD1010		错误步	发生操作错误的步 号码。	<ul style="list-style-type: none"> 当在应用指令的执行过程中发生操作错误时, 发生错误步的步号码, 被存储成 BIN 数据。此后, 每次发生操作错误, SD1010 中的内容都被更新。 	○

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后的 特殊寄存器	用于修改的 特殊寄存器	名称	含义	详细内容	相应的 CPU
D9011	SD1011		错误步	发生操作错误的步 号码。	● 当在应用指令的执行过程中发生操作错误时，发生 错误步的步号码，被存储成 BIN 数据。由于当 SM1011 从 OFF 变为 ON 时此步号码被存储到 SD1011， SD1011 中的数据并不被更新，除非 SM1011 被用户 程序清除。	○
D9014	SD1014		I/O 控制模式	I/O 控制模式号码	● I/O 控制模式设置返回到下列号码中的任何一个： 0. 输入和输出都处于直接模式 1. 输入处于刷新模式，输出处于直接模式 3. 输入和输出都处于刷新模式	
D9015	SD1015	SD203	CPU 的工作状态	CPU 的工作状态	● 如下所示的 CPU 的工作状态被存储到 SD203。 <div><div><div>b15 到 b12 b11 到 b8 b7 到 b4 b3 到 b0</div><div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div><div><div>通过计算机的远程 RUN/STOP</div><div><div>0</div><div>RUN</div></div><div><div>1</div><div>STOP</div></div><div><div>2</div><div>PAUSE *1</div></div></div><div><div>CPU钥匙开关</div><div><div>0</div><div>RUN</div></div><div><div>1</div><div>STOP</div></div><div><div>2</div><div>PAUSE *1</div></div><div><div>3</div><div>STEP RUN</div></div></div><div><div>保留和远程RUN/STOP 模式的一样</div></div><div><div>通过参数设置的远程 RUN/STOP</div><div><div>0</div><div>RUN</div></div><div><div>1</div><div>STOP</div></div><div><div>2</div><div>PAUSE *1</div></div></div></div><div><div>在程序中的状态</div><div><div>0</div><div>除下面以外的</div></div><div><div>1</div><div><div>STOP</div><div>指令执行</div></div></div></div><div>*1 当 CPU 模块处于 RUN 模式并且 SM1040 为 OFF 时， CPU 模块如果被改变到 PAUSE 模式，它也将保持为 RUN 模式。</div></div>	
D9016	SD1016		程序号码	0: 主程序 (ROM) 1: 主程序 (RAM) 2: 子程序 1 (RAM) 3: 子程序 2 (RAM) 4: 子程序 3 (RAM) 5: 子程序 1 (ROM) 6: 子程序 2 (ROM) 7: 子程序 3 (ROM) 8: 主程序 (E ² PROM) 9: 子程序 1 (E ² PROM) A: 子程序 2 (E ² PROM) B: 子程序 3 (E ² PROM)	● 表示当前运行的是哪个顺序程序。0 到 B 中的一个 值被以 BIN 码存储。	
D9017	SD1017	SD520	扫描时间	最小扫描时间 (10ms 为单位)	● 如果扫描时间小于 SD520 中的值，此值在每个 END 处理中被重新存储。也就是说，扫描时间的最小值被 以 BIN 码存储到 SD520。	
D9018	SD1018	SD524	扫描时间	扫描时间 (10ms 为单位)	● 扫描时间在每个 END 处理中以 BIN 码存储，并经常 被重写。	

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后的 特殊寄存器	用于修改的 特殊寄存器	名称	含义	详细内容	相应的 CPU
D9019	SD1019	SD526	扫描时间	最大扫描时间 (10ms 为单位)	● 如果扫描时间大于 SD526 中的值, 此值在每个 END 处理中被重新存储。也就是说, 扫描时间的最大值以 BIN 码存储到 SD526 中。	
D9020	SD1020		固定扫描	固定扫描时间 (用户设置以 10ms 为单位)	● 设置在 10ms 的倍数内启动的连续程序的间隔。 0: 无设置 1 到 200: 设置程序在 (设定值) × 10ms 的间隔上执行。	
D9021	SD1021	—	扫描时间	扫描时间 (1ms 为单位)	● 扫描时间以 BIN 码存储, 并在每个 END 之后被更新。	
D9022	SD1022	SD412	1 秒计数器	以 1s 为单位计数	● 当 PC CPU 开始运行时, 此寄存器开始每秒计数 1。 ● 寄存器开始从 0 到 32767 向上计数, 然后减小到 -32768 并重新向上计数到 0。计数重复此过程。	
D9025	SD1025	SD210	时钟数据	时钟数据 (年, 月)	● 以 BCD 码存储年(低 2 位数)和月。 	
D9026	SD1026	SD211	时钟数据	时钟数据 (日期, 小时)	● 以 BCD 码存储日期和小时。 	
D9027	SD1027	SD212	时钟数据	时钟数据 (分钟, 秒)	● 以 BCD 码存储分钟和秒。 	
D9028	SD1028	SD213	时钟数据	时钟数据 (星期几)	● 以 BCD 码存储星期几。 	
D9035	SD1035	SD648	扩展文件寄存器	使用的块号码	● 以 BCD 码存储正在使用的扩展文件寄存器的块号码。	
D9036	SD1036		设备号码的扩展 文件寄存器名称	当来自扩展文件寄存器的单个设备被直接访问时的设备号码。	● 以 BIN 数据在 SD1036 和 SD1037 上以 2 字指定用于扩展文件寄存器的直接读写的设备号码。使用从块号码 1 的 R0 开始到指定设备号码之间的连续号码。 	
D9037	SD1037					
D9038	SD1038	SD207	LED 显示优先级	优先级 1 到 4	● 设置 ERROR LED 的优先级, 它通过点亮(或闪烁), 用错误代码的号码来指示错误。 ● 优先级设置区域的组态如下所示。 	
D9039	SD1039	SD208		优先级 5 到 7		

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后的 特殊寄存器	用于修改的 特殊寄存器	名称	含义	详细内容	相应的 CPU
D9044	SD1044		用于采样跟踪	在采样跟踪过程中的步或时间	● 随外部设备变为 ON/OFF。 在扫描时 ----- 0 在时刻 ----- 时间(10ms 为单位) 以 BIN 码存储此值。	○
D9049	SD1049		用于 SFC 的工作区域	扩展文件寄存器的块号码	● 以二进制值存储扩展文件寄存器的块号码，此寄存器作为用于 SFC 程序执行的工作区域。 ● 存储“0”，如果存储一个 16K 或者更小的空白区域，此空白区域不能是第一个扩展文件寄存器，被使用或者 SM320 为 OFF。	
D9050	SD1050		SFC 程序错误号码	由 SFC 程序产生的错误代码	● 以 BIN 码存储在 SFC 程序中发生的错误的错误代码。 0： 无错误 80： SFC 程序参数错误 81： SFC 代码错误 82： 同时执行中超出的步数 83： 块启动错误 84： SFC 程序工作错误	
D9051	SD1051		错误块	发生错误处的块号码	● 以 BIN 码存储在 SFC 程序中发生错误代码 84 的步的号码。 ● 存储“0”，当错误代码 80, 81 或 82 发生时。 ● 当错误代码 83 发生时，存储块启动步的号码。	
D9052	SD1052			发生错误处的步号码	● 以 BIN 值存储在 SFC 程序中发生错误代码 84 的转换条件的号码。 ● 存储“0”，当错误代码 80, 81, 82 或 83 发生时。	
D9053	SD1053		错误转换	发生处错误的转换条件号码	● 以 BIN 码存储在 SFC 程序中发生错误代码 84 的转换条件的号码。 存储“0”，当错误代码 80, 81, 82 或 83 发生时。	
D9054	SD1054		错误顺序步	发生错误处的顺序步号码	● 以 BIN 码存储在 SFC 程序中发生错误代码 84 的转换条件和工作输出的顺序步的号码。	
D9055	SD1055	SD812	状态闭锁	状态闭锁步	● 存储状态闭锁被执行时的步号码。 ● 以二进制值存储当状态闭锁在主顺序程序中被执行时的步号码。 ● 如果状态闭锁在 SFC 程序中被执行，存储此块号码和步号码。 <div><div>块号码 (BIN)</div><div>步号码 (BIN)</div><div>← 高8位 → ← 低8位 →</div></div>	
D9060	SD1060	SD392	软件版本	内部软件的软件版本	● 以 ASCII 码存储内部系统的软件版本。 <div><div>高字节</div><div>低字节</div><div>↑</div></div> <div>在低字节处的数据是不确定的。 软件版本被存储在高字节的位置。 对于版本号“A”，举例来说，“41H”被存储</div> <div>注意：内部系统的软件版本可能和打印在包装盒后面的版本信息指示的版本号不一样。</div>	QnA
D9072	SD1072		PLC 通信检查	计算机链接数据检查	● 在串行通信模块的自我回路返回测试中，此串行通信模块通过自动读/写数据来进行通信检查。	○
D9081	SD1081	SD714	通信请求登记区域内的空白块数量	通信请求登记区域内的空白块数量	● 存储用于远程终端模块的通信请求登记区域内的空白块的数量，此远程终端模块连接到 MELSECNET/MINI-S3 主站单元，A2CCPU 或 A52GCPU。	QnA

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后的 特殊寄存器	用于修改的 特殊寄存器	名称	含义	详细内容	相应的 CPU
D9085	SD1085		用于设置时间检 查值的寄存器	默认值为 10s	<ul style="list-style-type: none">● 设置用于 MELSECNET/10 的数据链接指令 (ZNRD, ZNWR) 的时间检查时间。● 设置范围：1s 到 65535s(1 到 65535)● 设置单位：1s● 默认值：10s(如果 0 被设置, 则默认值 10s 将被使用)	
D9090	SD1090		额外的特殊功能 模块的数量	额外的特殊功能模 块的数量	<ul style="list-style-type: none">● 对于详细内容, 参考每个微型计算机程序包的手册。	
D9091	SD1091		详细错 误代码	自诊断的详细错 误 代码	<ul style="list-style-type: none">● 存储引起指令错误的详细代码。	
D9094	SD1094		被替换的 I/O 模 块的首 I/O 号码	被替换的 I/O 模块 的首 I/O 号码	<ul style="list-style-type: none">● 以 BIN 码存储 I/O 模块的首 I/O 号码的前两位数字, 此 I/O 模块将被在线拆除/安装。 例子) 输入模块 X2F0 → H2F	
D9100	SD1100	—	保险丝烧坏的模 块	以 16 数据点为单元 的位模式, 表示了 保险丝烧坏的模 块。	<ul style="list-style-type: none">● 保险丝烧坏的输出模块的号码 (以 16 数据点为单元) 被输入到位模式中。(当执行参数设置时, 预先设置输出单元号码。) <div><div>b15b14b13b12b11b10b9b8b7b6b5b4b3b2b1b0</div><div>SD110000001(YCD)00001(YBD)000000000000</div><div>SD1101000000000000000000000000</div><div>SD1107000000000000000000000000</div><div>↑</div><div>(表示烧坏的保险丝)</div></div>	
D9101	SD1101					
D9102	SD1102					
D9103	SD1103					
D9104	SD1104					
D9105	SD1105					
D9106	SD1106					
D9107	SD1107					
D9108	SD1108	—	步转换监视定时 器设置	定时器设置值和时间 溢出时 F 号码。	<ul style="list-style-type: none">● 设置步转换看门狗定时器的设定值, 以及当看门狗定时器时间到达时变为 ON 的报警器号码 (F 号码)。 <div><div>b15 到 b8b7 到 b0</div><div><div></div><div></div></div><div>定时器设置 (1到255s以秒为单)</div><div>F号码设置</div></div> <p>(通过将 MSM708 到 SM1114 中的任何一个变为 ON, 此监视定时器被启动。如果和此定时器相对应的步后面的转换条件在设定的时间内没有建立, 报警器 (F) 变为 ON。)</p>	
D9109	SD1109					
D9110	SD1110					
D9111	SD1111					
D9112	SD1112					
D9113	SD1113					
D9114	SD1114					
D9116	SD1116	—	I/O 模块校验错 误	以 16 数据点为单元 的位模式, 指示了 有块校验错误的模 块。	<ul style="list-style-type: none">● 当 I/O 模块, 它的数据和系统上电时输入的数据不同, 被检测到时, 此 I/O 单元的号码 (以 16 数据点为单元) 被输入到位模式。(在进行参数设置时预先设置 I/O 单元的号码。) <div><div>b15b14b13b12b11b10b9b8b7b6b5b4b3b2b1b0</div><div>SD1116000000000000000000000001(XY)</div><div>SD1117000000000000000000000000</div><div>SD1123000000(XY)0000000000000000</div><div>↑</div><div>(表示 I/O 模块校验错误)</div></div>	
D9117	SD1117					
D9118	SD1118					
D9119	SD1119					
D9120	SD1120					
D9121	SD1121					
D9122	SD1122					
D9123	SD1123					

特殊寄存器列表 (续表)

ACPU 特殊转换	转换后的 特殊寄存器	用于修改的 特殊寄存器	名称	含义	详细内容	相应的 CPU																																																																																																																																	
D9124	SD1124	SD63	报警器探测数量	报警器探测数量	<ul style="list-style-type: none">当 F0 到 255 (对于 AuA 和 AnU, 是 F0 到 2047) 中的一个被 SET F 变为 ON 时, 1 被加到 SD63 的值上。当 RST F 或 LEDR 指令被执行时, 1 被从 SD63 的值中减掉。 (如果 CPU 模块有 INDICATOR RESET 开关, 可以通过按下此开关来执行同样的处理。) 被 SET F 变为 ON 的数量被以 BIN 码存储进 SD63。SD63 的值最大为 8。	○																																																																																																																																	
D9125	SD1125	SD64	报警器探测号码	报警器探测号码	<ul style="list-style-type: none">当 F0 到 2047 中的任何一个被 SET F 变为 ON 时, 按照顺序变为 ON 的报警器号码 (F 号码) 被登记到 D9125 到 D9132 中。 被 RST F 关闭的 F 号码被从 D9125 到 D9132 中的一个擦除, 并且在被擦除的 F 号码移到前面的寄存器中之后, 这些 F 号码被存储。 通过执行 LEDR 指令, SD64 到 SD71 中的内容被向上移动一位。 (如果 CPU 模块有 INDICATOR RESET 开关, 可以通过按下此开关来执行同样的处理。) 当已经有 8 个报警器被探测到时, 第 9 个报警器即使已经被探测到也不被存储到 SD64 到 SD71 中。 <div>置位 置位 置位 RST 置位 置位 置位 置位 置位 置位 F50 F25 F99 F25 F15 F70 F65 F38 F110 F151 F210 LEDR</div>																																																																																																																																		
D9126	SD1126	SD65																																																																																																																																					
D9127	SD1127	SD66																																																																																																																																					
D9128	SD1128	SD67																																																																																																																																					
D9129	SD1129	SD68																																																																																																																																					
D9130	SD1130	SD69																																																																																																																																					
D9131	SD1131	SD70																																																																																																																																					
D9132	SD1132	SD71																																																																																																																																					
					<table><tr><td>SD62</td><td>0</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>99</td></tr><tr><td>SD63</td><td>0</td><td>1</td><td>2</td><td>3</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>8</td></tr><tr><td>SD64</td><td>0</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>99</td></tr><tr><td>SD65</td><td>0</td><td>0</td><td>25</td><td>25</td><td>99</td><td>99</td><td>99</td><td>99</td><td>99</td><td>99</td><td>99</td><td>15</td></tr><tr><td>SD66</td><td>0</td><td>0</td><td>0</td><td>99</td><td>0</td><td>15</td><td>15</td><td>15</td><td>15</td><td>15</td><td>15</td><td>70</td></tr><tr><td>SD67</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>70</td><td>70</td><td>70</td><td>70</td><td>70</td><td>65</td></tr><tr><td>SD68</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>65</td><td>65</td><td>65</td><td>65</td><td>38</td></tr><tr><td>SD69</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>38</td><td>38</td><td>38</td><td>110</td></tr><tr><td>SD70</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>110</td><td>110</td><td>151</td></tr><tr><td>SD71</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>151</td><td>210</td></tr></table>		SD62	0	50	50	50	50	50	50	50	50	50	50	99	SD63	0	1	2	3	2	3	4	5	6	7	8	8	SD64	0	50	50	50	50	50	50	50	50	50	50	99	SD65	0	0	25	25	99	99	99	99	99	99	99	15	SD66	0	0	0	99	0	15	15	15	15	15	15	70	SD67	0	0	0	0	0	0	70	70	70	70	70	65	SD68	0	0	0	0	0	0	0	65	65	65	65	38	SD69	0	0	0	0	0	0	0	0	38	38	38	110	SD70	0	0	0	0	0	0	0	0	0	110	110	151	SD71	0	0	0	0	0	0	0	0	0	0	151
SD62	0	50	50	50	50	50	50	50	50	50	50	99																																																																																																																											
SD63	0	1	2	3	2	3	4	5	6	7	8	8																																																																																																																											
SD64	0	50	50	50	50	50	50	50	50	50	50	99																																																																																																																											
SD65	0	0	25	25	99	99	99	99	99	99	99	15																																																																																																																											
SD66	0	0	0	99	0	15	15	15	15	15	15	70																																																																																																																											
SD67	0	0	0	0	0	0	70	70	70	70	70	65																																																																																																																											
SD68	0	0	0	0	0	0	0	65	65	65	65	38																																																																																																																											
SD69	0	0	0	0	0	0	0	0	38	38	38	110																																																																																																																											
SD70	0	0	0	0	0	0	0	0	0	110	110	151																																																																																																																											
SD71	0	0	0	0	0	0	0	0	0	0	151	210																																																																																																																											

(8) 保险丝烧坏的模块

号码	名称	含义	解释	设定者 (当被设 定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD1300	保险丝烧坏模 块	以 16 数据点为 单元的位模式。表 示了保险丝烧坏的 模块 0: 无烧坏的保险 丝 1: 有烧坏的保险 丝	<ul style="list-style-type: none">● 保险丝烧坏的输出模块的数量被以位模式输入 (以 16 数据点为单元的)。 (如果模块号码由参数设置, 此参数设置的号码被存储)● 对于远程站点的输出模块, 也检查保险丝烧坏的情况。 <div><div>b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0</div><div>SD1300<div>0001000100000000</div></div><div>SD1301<div>1000010000000000</div></div><div>SD1331<div>0000100000000100</div></div></div> <div>表示烧坏的保险丝</div> <ul style="list-style-type: none">● 即使烧坏的保险丝被新的替换掉, 寄存器也不被清除。 此标志位由错误复位操作清除。	S (错误)	D9100	○+Rem
SD1301					D9101	
SD1302					D9102	
SD1303					D9103	
SD1304					D9104	
SD1305					D9105	
SD1306					D9106	
SD1307					D9107	
SD1308					新增	
SD1309 to SD1330					新增到新增	
SD1331					新增	
SD1350 to SD1381	外部电源供电 分离模块 (用于将来的 扩展)	以 16 数据点为单 元的位模式, 表 示了外部电源供电 被分离的模块 0: 外部电源供电 分离 1: 外部电源供电 未分离	外部电源供电被分离的模块的号码 (以 16 数据点为单元的) 以位模式输入。 (如果此模块号码由参数设置, 此参数设置的号码被使用。)	S (错误)	新增	
			<div><div>b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0</div><div>SD1350<div>0001000100000000</div></div><div>SD1351<div>1000010000000000</div></div><div>SD1381<div>0000100000000100</div></div></div> <div>表示烧坏的保险丝</div>			

特殊寄存器列表 (续表)

(9) I/O 模块校验

号码	名称	含义	解释	设定者 (当被设定 定时)	相应的 ACPU D9 <div><div></div><div></div><div></div></div>	相应的 CPU
SD1400	I/O 模块校验 错误	以 16 数据点为单 元的位模式，表示 了有校验错误的模 块。 0: 无 I/O 校验错误 1: 有 I/O 校验错误	<div><div>● 当 I/O 模块信息和系统上电时登 记的信息不同的 I/O 模块被探 测到时，这些 I/O 模块的号码以位模式输入。 (如果模块号码由参数设置，此参数设置的号码被存储。)</div><div>● 同时检查 I/O 模块信息</div></div> <div><div><div><div><div>b15</div><div>b14</div><div>b13</div><div>b12</div><div>b11</div><div>b10</div><div>b9</div><div>b8</div><div>b7</div><div>b6</div><div>b5</div><div>b4</div><div>b3</div><div>b2</div><div>b1</div><div>b0</div></div><div><div><div>D9116</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div><div>1</div><div>($\frac{1}{16}$)</div></div></div><div><div><div>D9117</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div><div>1</div><div>($\frac{2}{16}$)</div></div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div><div><div><div>D9123</div><div><div><div>1</div><div>($\frac{1}{16}$)</div></div></div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div></div></div><div><div>表示一个 I/O 模块校验错误</div></div></div><div><div>● 即使烧坏的保险丝 被新的替换掉，寄 存器也不被清除。 此标志位由错误复位操作清除。</div></div></div><td>S (错误)</td><td><div>D9116</div><div>D9117</div><div>D9118</div><div>D9119</div><div>D9120</div><div>D9121</div><div>D9122</div><div>D9123</div><div>新增</div><div><div>新增到新增</div><div>新增</div></div></td><td><div>○ +Rem</div></td></div></div></div>	S (错误)	<div>D9116</div> <div>D9117</div> <div>D9118</div> <div>D9119</div> <div>D9120</div> <div>D9121</div> <div>D9122</div> <div>D9123</div> <div>新增</div> <div><div>新增到新增</div><div>新增</div></div>	<div>○ +Rem</div>

(10) 过程控制指令

号码	名称	含义	解释	设定者 (当被设定 定时)	相应的 ACPU D9 <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	相应的 CPU
SD1500 SD1501	基本周期	基本周期	<ul style="list-style-type: none">设置用于使用浮点数数据的过程控制指令的基本周期(1 秒为单位) <p>浮点数数据 = <input type="text" value="SD1501"/> <input type="text" value="SD1500"/></p>	U	新增	Q4AR
SD1502	过程控制指令详细错误代码	过程控制指令详细错误代码	<ul style="list-style-type: none">显示了在过程控制指令中发生的错误的详细错误内容。	S (发生错误)	新增	
SD1503	过程控制指令产生的错误的定位	过程控制指令产生的错误的定位	<ul style="list-style-type: none">显示了在过程控制指令发生的错误的错误过程块。	S (发生错误)	新增	
SD1506 SD1507	空设备	空设备	<ul style="list-style-type: none">被过程控制指令用于指定空设备。	U	新增	

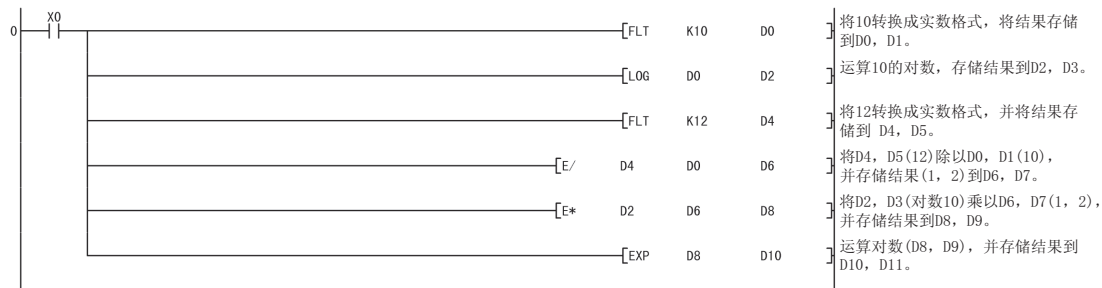
附录 5 应用程序实例

附录 5.1 执行 x^n , $\sqrt[n]{x}$ 操作的程序概念

虽然 QCPU 和 QnACPU 没有执行 x^n , $\sqrt[n]{x}$ 操作的指令, 但是 LOG 和 EXP 可以用在一起去执行 x^n , $\sqrt[n]{x}$ 操作。

(1) 执行 x^n 操作的程序的概念可以通过使用 $e^{(n \log_e x)}$ 来实现。

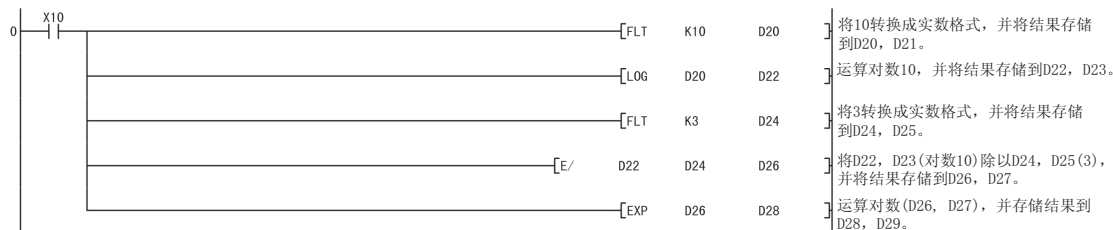
举例来说, 101.2 的操作就是 $e^{(1.25 \log_e 10)}$, 它在顺控程序中的表达方式如下所示。



(2) 执行 $\sqrt[n]{x}$

$\sqrt[n]{x}$ 操作的程序的概念可以通过使用 $e^{(\frac{1}{n} \log_e x)}$ 来实现。

举例来说, $\sqrt[3]{10}$ 的操作就是 $e^{(\frac{1}{3} \times \log_e 10)}$, 它在顺控程序中的表达方式如下所示。



索引

[A]

- A 触点并行连接 (OR) 5-2
- A 触点操作启动 (LD) 5-2
- A 触点串行连接 (AND) 5-2
- ACOS(浮点数据的 \cos^{-1} 运算) 7-250
- ADRSET(直接地址读) 7-333
- ANB(梯形图块系列) 5-7
- AND(A 触点串行连接) 5-2
- AND[=, <>, <, >=, >, <=]
(BIN 16 位数据比较) 6-2
- AND\$[=, <>, <, >=, >, <=]
(字符串数据比较) 6-8
- ANDD[=, <>, <, >=, >, <=]
(BIN 32 位数据比较) 6-4
- ANDE[=, <>, <, >=, >, <=]
(浮点数数据比较) 6-6
- ANDF(脉冲串行连接 • 上升沿) 5-5
- ANDP(脉冲串行连接 • 下降沿) 5-5
- ANI(B 触点串行连接) 5-2
- ASC(十六进制 BIN 到 ASCII 的转换) 7-225
- ASCII 码 LED 显示指令 (LED) 7-154
- ASIN(浮点数据的 \sin^{-1} 运算) 7-248
- ATAN(浮点数据的 \tan^{-1} 运算) 7-252
- ASCII 到十六进制 BIN 的转换 (HEX) 7-227

[B]

- B-(减 BCD 4 位数据) 6-28
6-30
- B 触点操作启动 (LDI) 5-2
- B*(乘 BCD 4 位数据) 6-36
- B*, B/(BCD 4 位数据加法和减法运算) 6-36
- B/(除 BCD 4 位数据) 6-36
- B+(加 BCD 4 位数据) 6-28
6-30
- BACOS(BCD 类型 \cos^{-1} 运算) 7-277
- BAND(16 位死区控制) 7-284
- BASIN(BCD 类型 \sin^{-1} 运算) 7-275
- BATAN(BCD 类型 \tan^{-1} 运算) 7-279
- BCD(BIN 数据到 4 位数) 6-57
- BCD(BIN 16 位数到 BCD 4 位数的转换) 6-57
- BCD 格式数据到浮点数 (EREXP) 7-240
- BCD 4 位数据乘法和除法运算 (B*, B/) 6-36
- BCD 4 位数据到 BIN 数据 (BIN) 6-59
- BCD 4 位数据加法和减法运算 (B+, B-) 6-28
6-30
- BCD 4 位数据平方根 (BSQR) 7-266

- BCD 4 位数据到 BIN 16 位的转换 (BIN) 6-59
- BCD 4 位数据到十进制
ASCII 的转换 (BCDDA) 7-185
- BCD 8 位数据乘法和除法运算 (DB*, DB/) 6-38
- BCD 8 位数据加法和减法运算 (DB+, DB-) 6-32
6-34
- BCD 8 位数据到 BIN 32 位的转换 (DBIN) 6-59
- BCD 8 位数据到十进制 ASCII 的转换
(DBCDDA) 7-185
- BCD 8 位数据平方根 (BDSQR) 7-266
- BCDDA(BCD 4 位数据到十进制 ASCII 的转换) 7-185
- BCD 的转换
BIN 16 位到 BCD 4 位数据的转换 (BCD) 6-57
- BIN 32 位到 BCD 8 位数据的转换 (DBCD) 6-57
- BCD 类型 \cos^{-1} 运算 (BACOS) 7-277
- BCD 类型 COS 运算 (BCOS) 7-271
- BCD 类型 \sin^{-1} 运算 (BASIN) 7-275
- BCD 类型 SIN 运算 (BSIN) 7-269
- BCD 类型 \tan^{-1} 运算 (BATAN) 7-279
- BCD 类型 TAN 运算 (BTAN) 7-273
- BCOS(BCD 类型 COS 运算) 7-271
- BDSQR(BCD 8 位数据平方根) 7-266
- BIN(BCD 4 位数据到 BIN 数据) 6-59
- BIN(BCD 4 位数据到 BIN 16 位的转换) 6-59
- BIN 16 位到 BIN 32 位的转换 (DBL) 6-65
- BIN 16 位到浮点数据的转换 (FLT) 6-61
- BIN 16 位到格雷码的转换 (GRY) 6-67
- BIN 16 位到十进制 ASCII 的转换 (BINDA) ... 7-179
- BIN 16 位到十六进制 ASCII 的转换 (BINHA) . 7-182
- BIN 16 位到字符串的转换 (STR) 7-203
- BIN 16 位乘法和除法运算 (*, /) 6-24
- BIN 16 位到 BIN 32 位 (字) 6-66
- BIN 16 位的最高和最低限控制 (LIMIT) 7-281
- BIN 16 位加法和减法运算 (+, -) 6-16
6-18
- BIN 16 位数据 (NEG) 的 2 位补码 6-71
- BIN 16 位数据比较
([=, <>, <, >=, >, <=]) 6-2
- BIN 16 位数据的零区控制 (ZONE) 7-287
- BIN 16 位数据排序 (SORT) 7-80
- BIN 32 位到浮点数的转换 (DFLT) 6-61
- BIN 32 位到格雷码的转换 (DGRY) 6-67
- BIN 32 位到十进制
ASCII 的转换 (DBINDA) 7-179
- BIN 32 位到十六进制
ASCII 的转换 (DBINHA) 7-182

变址修改界限	3-11
变址修改中修正值的指定 (IXDEV, IXSET) ...	7-126
并行连接 (OR, ORI) ...	5-2
步数	3-20
[C]	
CALL (子程序调用) ...	7-91
CGMODE (CPU 切换时间运算模式设置) ...	10-4
CHKCIR (改变 CHK 指令的检查格式) ...	7-165
CHKEND (改变 CHK 指令的检查格式) ...	7-166
CHKST, CHK (特殊格式故障检查) ...	7-161
CJ (指针分支) ...	6-96
CML (16 位负转移) ...	6-84
COM (刷新指令) ...	7-112 9-50
COMRD (读软元件注释数据) ...	7-196
COS (浮点数据的 COS 运算) ...	7-244
操作错误	3-16
操作复位脉冲-被边沿继电器存储在内存中 (EGF, EGP) ...	5-16
操作复位脉冲-存储在内存中	5-13
操作结果出栈 (MPS) ...	5-9
操作结果读 (MRD) ...	5-9
操作结果取反 (INV) ...	5-13
操作结果入栈 (MPP) ...	5-9
操作启动 (LD, LDI) ...	5-2
乘法	
BCD 4 位数据 (B*) ...	6-36
BCD 8 位数据 (DB*) ...	6-38
BIN 16 位 (*) ...	6-24
BIN 32 位 (D*) ...	6-26
浮点数据 (E*) ...	6-44
程序备用指令 (PSTOP) ...	7-315
程序低速执行登记 (PLOW) ...	7-320
程序跟踪 (PTRA) ...	7-177
程序码指令	2-22
程序扫描执行登记指令 (PSCAN) ...	7-318
程序输出 OFF 备用指令 (POFF) ...	7-316
程序执行控制指令	2-22
程序指令	2-43
除法	
BCD 4 位数据 (B/) ...	6-36
BCD 8 位数据 (DB/) ...	6-39
BIN 16 位 (/) ...	6-25
BIN 32 位 (D/) ...	6-27
浮点数据 (E/) ...	6-45
触点的脉冲转换 (DELTA) ...	5-38
触点指令	2-

触点指令

并行连接(OR, ORI)	5-2
操作启动(LD, LDI)	5-2
串行连接(AND, ANI)	5-2
脉冲并行连接(ORF, ORP)	5-5
脉冲串行连接(ANDF, ANDP)	5-5
脉冲运算启动(LDF, LDP)	5-5
从数据表读最旧的数据(FIFR)	7-133
从数据表读最新的数据(FPOP)	7-135
从数据表中删除数据(FDEL)	7-137
从文件寄存器中读 1 个字节(ZRRDB)	7-329
从远程 I/O 站点上的特殊功能模块	
中读数据(ZNFR)	8-67
从远程 I/O 站点特殊功能模块中读数据(RFRP)	8-91
从中断程序中恢复(IRET)	6-109
从子程序中返回(RET)	7-95
从左边提取字符串数据(LEFT)	7-229
错误显示和报警器复位指令(LEDRL)	7-158

[D]

D-(BIN 32 位数据减法运算)	6-20
.....	6-22
D*(BIN 32 位数据乘法运算)	6-26
D/(BIN 32 位数据除法运算)	6-27
D[=, <>, <, >=, >, <=]	
(BIN 32 位数据比较)	6-4
D+(BIN 32 位数据加法运算)	6-20 6-22
DABCD(十进制 ASCII 到 BCD 4 位数的转换) ..	7-193
DABIN(十进制 ASCII 到 BIN 16 位的转换) ..	7-191
DAND(32 位数据的逻辑乘积)	7-3 7-6
DATE-(时钟数据减法运算)	7-306
DATE+(时钟数据加法运算)	7-304
DATERD(读时钟数据)	7-296
DATEWR(写时钟数据)	7-300
DB-(减 BCD 8 位数据)	6-32 6-34
DB*(乘 BCD 8 位数据)	6-38
DB/(除 8 位数据 BCD 数据)	6-39
DB+(加 BCD 8 位数据)	6-32
.....	6-34
DBAND(32 位死区控制)	7-284
DBCD(BIN 32 位到 BCD 8 位数的转换)	6-57
DBCD(BIN 到 BCD 8 位数的转换)	6-57
DBCDDA(BCD 8 位数据到十进制	
ASCII 的转换)	7-185
DBIN(BCD 8 位数据到 BIN 32 位的转换)	6-59
DBINDA(BIN 32 位到十进制	
ASCII 的转换)	7-179
DBINHA(BIN 32 位到十六进制	
ASCII 的转换)	7-182

DBL(BIN 16 位到 BIN 32 位的转换)	6-65
DCML(32 位负转换)	6-84
DDABCD(十进制 ASCII 到	
BCD 8 位数的转换)	7-193
DDABIN(十进制 ASCII 到 BIN 32 位的转换) ..	7-188
DDEC(32 位 BIN 数据递减)	6-55
DEC(16 位 BIN 数据递减)	6-53
DECO(8 位到 256 位解码)	7-56
DEG(浮点数弧度到角度的转换)	7-256
DELTA(直接输出的脉冲转换)	5-38
DFLT(BIN 32 位到浮点数的转换)	6-61
DFRO(特殊功能模块 2 字数据读运算)	7-140
DGBIN(格雷码到 BIN 32 位的转换)	6-69
DGRY(BIN 32 位到格雷码的转换)	6-67
DHABIN(十六进制 ASCII 到	
BIN 32 位的转换)	7-191
DI(中断禁止)	6-100
DINC(递增 32 位 BIN)	6-55
DINT(浮点数据到 BIN 32 位的转换)	6-63
DIS(16 位数据的 4 位编组)	7-62
DLIMIT(BIN 32 位数据的上限和下限控制) ..	7-281
DMAX(32 位数据的最大值搜索)	7-75
DMIN(32 位数据的最小值搜索)	7-77
DMOV(32 位转移)	6-78
DNEG(BIN 32 位数据的 2 位补码)	6-71
DOR(32 位数据的逻辑和)	7-10
.....	7-12
DRCL(32 位数据的左循环)	7-36
DRCR(32 位数据的右循环)	7-34
DROL(32 位数据的左循环)	7-36
DROR(32 位数据的右循环)	7-34
DSER(32 位数据搜索)	7-50
DSFL(n 字数据左移 1 字)	7-42
DSFR(n 字数据右移 1 字)	7-42
DSORT(BIN 32 位数据排序)	7-80
DSTR(BIN 32 位到字符串的转换)	7-203
DSUM(32 位数据检查)	7-54
DTEST(位测试)	7-46
DTO(特殊功能模块 2 字数据写运算)	7-143
DUTY(定时脉冲的产生)	7-326
DVAL(字符串到 BIN 32 位的转换)	7-209
DWSUM(32 位数据的总数计算)	7-85
DXCH(16 位数据交换)	6-91
DXNR(32 位数据否定排它逻辑求和运算)	7-22
.....	7-26
DXOR(32 位排它或运算)	7-16
.....	7-18
DZONE(BIN 32 位数据的零区控制)	7-287
打印 ASCII 码指令(PR)	7-146

打印注释指令 (PRC)	7-149	FLT (BIN 16 位到浮点数的转换)	6-61
低速保持定时器 (OUT ST)	5-20	FMOV (16 位数据块转移)	6-89
低速定时器 (OUT T)	5-20	FOR (FOR 到 NEXT)	7-87
递减		FOR 到 NEXT (FOR)	7-87
16 位 BIN (DEC)	6-53	FOR 到 NEXT 指令循环的强制结束 (BREAK)	7-89
32 位 BIN (DDEC)	6-55	FPOP (从数据表读取新数据)	7-135
递增 16 位 BIN (INC)	6-53	FREAD (从指定文件中读取数据)	9-16
递增 32 位 BIN (DINC)	6-55	FROM (其它站点的 CPU 共享存储区读取数据) ..	9-48
定时脉冲产生 (DUTY)	7-326	FROM (特殊功能模块 1 字数据读运算)	7-140
定时器 (OUT T)	5-20	FWRITE (写数据到指定的文件)	9-8
读路径参数 (RTREAD)	8-99	发送数据到其它站点 (SEND)	8-41
读软元件注释数据 (COMRD)	7-196	浮点数的符号取反 (ENEG)	6-73
读时钟数据 (DETARD)	7-296	浮点数到字符串的转换 (ESTR)	7-214
[E]		浮点数弧度到角度的转换 (DEG)	7-256
E- (减浮点数)	6-41	浮点数角度到弧度的转换 (RAD)	7-254
	6-43	浮点数据到 BIN 16 位的转换 (INT)	6-63
E* (乘以浮点数)	6-44	浮点数据到 BIN 32 位的转换 (DINT)	6-63
E/ (除以浮点数)	6-45	浮点数据 COS-1 运算 (ACOS)	7-250
E[, <>, <, >=, >, <=] (比较实数数据)	6-6	浮点数据 COS 运算 (COS)	7-244
E[, <>, <, >=, >, <=]		浮点数据比较	6-6
(浮点数据比较)	6-6	浮点数据加法和减法运算 (E+, E-)	6-40
E+ (加浮点数)	6-40	浮点数据乘法和除法 (E*, E/)	6-44
	6-42	浮点数据的 SIN ⁻¹ 运算 (ASIN)	7-248
ECALL (在程序文件之间的子程序调用)	7-100	浮点数据的 TAN ⁻¹ 运算 (ATAN)	7-252
EEPROM 文件寄存器的批量写操作 (EROMWR) ...	7-342	浮点数据的自然对数运算 (LOG)	7-262
EFCALL (在程序文件之间的输出 OFF 调用) ...	7-104	浮点数据转移 (EMOV)	6-80
EGF (脉冲运算结果 • 上升沿)	5-16	浮点数区域的平方根运算 (SQR)	7-258
EGP (脉冲运算结果 • 下降沿)	5-16	浮点数数据的 SIN 运算 (SIN)	7-242
EI (中断允许)	6-100	浮点数数据的 TAN 运算 (TAN)	7-246
EMOD (浮点数到 BCD)	7-238	浮点数数据的指数运算 (EXP)	7-260
EMOV (浮点数据转移)	6-80	复位 (RST)	5-30
ENCO (256 到 8 位编码)	7-58	复位报警器 (RST F)	5-32
END (结束顺控程序)	5-48	复位采样跟踪 (STRAR)	7-175
ENEG (浮点数的符号取反)	6-73	复位程序跟踪 (PTRAR)	7-177
EREXP (BCD 格式数据到浮点数)	7-240	复位电机控制 (MCR)	5-42
EROMWR (批量写运算到 EEPROM 文件寄存器) ..	7-342	复位看门狗定时器 (WDT)	7-324
ESTR (浮点数到字符串的转换)	7-214	复位状态闭锁 (SLTR)	7-173
EVAL (字符串到浮点数的转换)	7-221	[G]	
EXP (浮点数数据的指数运算)	7-260	GBIN (格雷码到 BIN 16 位的转换)	6-69
[F]		GOEND (跳转到 END)	6-99
FCALL (输出 OFF 调用)	7-100	GRY (BIN 16 位到格雷码的转换)	6-67
FDEL (从数据表中删除数据)	7-137	改变 CHK 指令的检查格式 (CHKCIR, CHKEND) .	7-165
FEND (结束主程序)	5-46	改变时间数据格式 (HOUR)	7-308
FF (位软元件输出取反)	5-36	改变时间数据格式 (SECOND)	7-308
FIFR (从数据表读取旧数据)	7-133	高速保持定时器 (OUTH ST)	5-20
FIFW (写数据到数据表)	7-131	高速定时器 (OUTH T)	5-20
FINS (在数据表中插入数据)	7-137	格雷码到 BIN 16 位 (GBIN)	6-69
		格雷码到 BIN 16 位数据的转换 (GBIN)	6-69

格雷码到 BIN 32 位 (DGBIN)	6-69	浮点数据 (E+)	6-40
格雷码到 BIN 32 位数据的转换 (DGBIN)	6-69		6-42
[H]		块 (BK+)	6-46
HABIN (十六进制 ASCII 到			6-42
BIN 16 位的转换)	7-191	间接指定	3-13
HEX (ASCII 到十六进制 BIN 的转换)	7-227	减法	
HOUR (改变时间数据格式)	7-308	BCD 4 位数据 (B-)	6-28
和远程 I/O 站点通讯			6-30
从特殊功能模块读数据 (RFRP)	8-91	BCD 8 位数据 (DB-)	6-32
从特殊功能模块读数据 (ZNFR)	8-67		6-35
写数据到特殊功能模块 (RTOP)	8-95	BIN 16 位 (-)	6-17
写数据到特殊功能模块 (ZNT0)	8-72		6-19
缓冲存储区访问指令	2-33	BIN 32 位 (D-)	6-21
或 (OR)	5-2		6-23
或取反 (ORI)	5-2	浮点数据 (E-)	6-40
			6-42
[I]		块 (BK-)	6-46
I/O 刷新	2-22	教学定时器 (TTMR)	6-117
I/O 刷新 (RFS)	6-111	结构体创建指令	2-31
I/O 刷新指令	2-22	结束顺控程序 (END)	5-48
IMASK (中断程序掩码)	6-100	结束主程序 (FEND)	5-46
INC (16 位 BIN 数据递减)	6-53	矩阵输入 (MTR)	6-131
INC (16 位 BIN 数据递增)	6-53	[K]	
INSTR (字符串搜索)	7-236	KEY (来自键盘的数字键输入)	7-334
INT (浮点数据到		空操作 (NOP, NOPLF, PAGE)	5-52
BIN 16 位的转换)	6-63	块 BCD 4 位数据到块	
INV (操作结果取反)	5-13	BIN 16 位数据的转换 (BKBIN)	6-76
IRET (从中断程序中恢复)	6-118	块 16 位交换 (BXCH)	6-93
IX, IXEND (整个梯形图的变址修改)	7-118	块 16 位转移 (BMOV)	6-87
IXDEV (变址修改中修正值的指定)	7-126	块否定排它逻辑求和运算 (BKXNR)	7-28
IXSET (变址修改中修正值的指定)	7-126	块 BIN 16 位数据到	
[J]		BCD 4 位数据的转换 (BKBCD)	6-74
JMP (指针分支)	6-96	块加法 (BK+)	6-46
将消息显示到外围软元件 (MSG)	7-310	块减法 (BK-)	6-47
基本指令	2-8	块逻辑乘 (BKAND)	7-8
计数器 (OUT C)	5-24	块逻辑和运算 (BKOR)	7-14
计数器 1 相输入增大或减小 (UDCNT1)	6-113	块排它或运算 (BKXOR)	7-20
计数器 2 相输入增大或减小 (UDCNT2)	6-115	[L]	
加法		LD\$ [=, <>, <, >=, >, <=]	
BCD 4 位数据 (B+)	6-28	(字符串数据比较)	6-8
	6-30	LD (A 触点运算启动)	5-2
BCD 8 位数据 (DB+)	6-32	LD [=, <>, <, >=, >, <=]	
	6-34	(BIN 16 位数据比较)	6-2
BIN 16 位 (+)	6-16	LDD [=, <>, <, >=, >, <=]	
	6-18	(BIN 32 位数据比较)	6-4
BIN 32 位 (D+)	6-20	LDE [=, <>, <, >=, >, <=]	
	6-22	(浮点数据比较)	6-6

LDF(脉冲运算开始 • 下降沿)	5-5	[N]	
LDI (B 触点运算开始)	5-2	NDIS(随机数据的分解)	7-66
LDP(脉冲运算开始 • 上升沿)	5-5	NEG(BIN 16 位数据的 2 位补码)	6-71
LED(ASCII 码 LED 显示指令)	7-154	NEXT(到 NEXT)	7-87
LEDC(用于注释的 LED 显示指令)	7-156	NOP(空操作)	5-52
LEDR(错误显示和报警器复位指令)	7-158	NOPLF(无操作页改变)	5-52
LEFT(左边提取字符串数据)	7-229	NUNI(随机数据的连接)	7-66
LEN(字符串长度检测)	7-201	n 位数据的 1 位右移(BSFR)	7-40
LIMIT(最高和最低限控制 BIN 16 位)	7-281	n 位数据的 1 位左移(BSFL)	7-40
LOG(浮点数数据的自然对数运算)	7-262	n 字数据的 1 字右移(DSFR)	7-42
来自键盘的数字键输入(KEY)	7-334	n 字数据的 1 字左移(DSFL)	7-42
来自其它站点的瞬时请求(REQ)	8-55		
来自外部软元件的键盘输入(PKEY)	7-312	[O]	
连接随机数据(NUNI)	7-66	OR\$[=, <>, <, >=, >, <]	
连接指令	2-5	(字符串数据比较)	6-8
连接指令链接字符串	6-49	OR(A 触点并行连接)	5-3
连接指令梯形图块并行连接(ORB)	5-7	OR[=, <>, <, >=, >, <=]	
连接指令梯形图块串行连接(ANB)	5-7	(BIN 16 位数据比较)	6-2
连接字符串(\$+)	6-49	ORB(梯形图块并行连接)	5-7
逻辑运算指令	2-24	ORD[=, <>, <, >=, >, <=]	
		(BIN 32 位数据比较)	6-4
[M]		ORE[=, <>, <, >=, >, <=]	
MAX(16 位数据的最大值搜索)	7-75	(浮点数据比较)	6-6
MC(置位电机控制)	5-42	ORF(脉冲并行连接 • 下降沿)	5-6
MCR(复位电机控制)	5-42	ORI(B 触点并行连接)	5-2
MEF(脉冲运算结果下降沿)	5-14	ORP(脉冲并行连接 • 上升沿)	5-6
MEP(脉冲运算结果上升沿)	5-14	OUT	
MIDR(字符串中随机选择)	7-232	低速保持定时器(OUTH ST)	5-20
MIDW(字符串中随机选择替换)	7-232	低速定时器(OUTH T)	5-20
MIN(16 位数据的最小值搜索)	7-77	高速定时器(OUTH T)	5-20
MOV(16 位转移)	6-78	高速累计定时器(OUTH ST)	5-20
MPP(操作结果入栈)	5-9	计数器(OUT C)	5-24
MPS(操作结果出栈)	5-9		
MRD(操作结果读)	5-9	[P]	
MSG(消息显示到外围软元件)	7-310	PAGE(无操作页改变)	5-52
MTR(矩阵输入)	6-132	PKEY(来自外围软元件的键盘输入)	7-312
脉冲(PLS)	5-34	PLF(下降沿输出)	5-34
脉冲并行连接(ORF, ORP)	5-5	PLOADP(存储器中载入程序)	9-30
脉冲串行连接(ANDF, ANDP)	5-5	PLOW(程序低速执行登记)	7-320
脉冲宽度调制(PWM)	6-130	PLS(上升沿输出)	5-34
脉冲密度(SPD)	6-126	PLSY(固定周期脉冲输出)	6-128
脉冲输出(PLSY)	6-128	POFF(程序输出 OFF 备用指令)	7-316
脉冲运算结果 • 下降沿(EGP)	5-16	Pop(MPP)	5-9
脉冲运算结果 • 上升沿(EGF)	5-16	PR(打印 ASCII 码指令)	7-146
脉冲运算开始	5-5	PRC(打印注释指令)	7-149
脉冲转换(EGF, EGP)	5-16	PSCAN(程序扫描执行登记指令)	7-318
脉冲转换(MEF, MEP)	5-14	PSTOP(程序备用指令)	7-315

PSWAPP(装载+卸载)	9-31	RTOP(写数据到远程 I/O 站点的特殊 功能模块)	8-95
PTRA(程序跟踪的设置)	7-177	RTREAD(读路径参数)	8-99
PTRAEXE(程序跟踪的执行)	7-179	RTWRITE(写路径参数)	8-103
PTRAR(程序跟踪的复位)	7-177	如何阅读指令表	2-2
PUNLOADP(程序存储器中卸载程序)	9-29	入栈(MPP)	5-9
PWM(脉冲宽度调制)	6-130	软元件范围检查	3-16
[Q]		软元件数据写进站点(WRITE)	8-27
QCDSET(用于注释的文件设置)	7-294	[S]	
QDRSET(用于文件寄存器使用的设置文件) ...	7-292	S.T0(写到主机站点的 CPU 共享存储器上)	9-41
其他站读软元件数据(ZNRD)	8-77	SCJ(指针分支)	6-96
其他站读字软元件数据(READ)	8-15	SECOND(改变时间数据格式)	7-308
其他站读字软元件数据(SREAD)	8-21	SEG(7-段解码)	7-60
其他站接收数据(RECV)	8-49	SEND(发送数据到其它站点)	8-41
其它使用方便的指令	2-23	SER(16 位数据搜索)	7-50
其它指令	5-50	SET-设置	5-28
	7-324	SET-设置报警器(SET F)	5-32
嵌套	5-46	SET-设置软元件(SET)	5-28
切换文件寄存器号码(RSET)	7-290	SFL(16 位数据的 n 位左移)	7-38
切换指令	2-41	SFR(16 位数据的 n 位右移)	7-38
	7-290	SFT(位软元件移位)	5-40
取反		SIN(浮点数数据的 SIN 运算)	7-242
操作结果(INV)	5-13	SLT(设置状态闭锁)	7-173
浮点数符号(ENEG)	6-73	SLTR(复位状态闭锁)	7-173
位软元件输出(FF)	5-36	SORT(BIN 16 位数据排序)	7-80
[R]		SPD(脉冲密度测量)	6-126
RAD(浮点数角度到弧度的转换)	7-254	SPREF(缓冲存储器批量刷新)	10-10
RAMP(斜坡信号)	6-124	SQR(浮点数区域的平方根运算)	7-258
RBMOV(文件寄存器的高速块转移)	9-36	SREAD(其它站点读字软元件数据)	8-21
RCL(16 位数据的循环左移)	7-32	SRND(随机数生成和序列更新)	7-264
RCR(16 位数据的循环右移)	7-30	STMODE(CPU 启动过程中的操作模式 设置指令)	10-2
Read(MRD)	5-9	STMR(特殊定时器)	6-119
READ(其它站点读字软元件数据)	8-13	STOP(顺控程序停止)	5-50
RECV(其它站点接收数据)	8-49	STR(BIN 16 位到字符串的转换)	7-203
REQ(来自其它站点的暂时请求)	8-55	STRA(设置采样跟踪)	7-175
RET(子程序中返回)	7-95	STRAR(复位采样跟踪)	7-175
RFRP(远程 I/O 站点特殊功能 模块中读数据)	8-91	SUM(16 位数据检查)	7-54
RFS(I/O 刷新)	6-111	SWAP(最高和最低字节交换)	6-95
RIGHT(右边提取字符串数据右边)	7-229	SWRITE(写软元件数据到其它站点)	8-34
RND(随机数生成和序列更新)	7-264	设定值的变址修改	3-11
ROL(16 位数据的循环左移)	7-32	设置采样跟踪(STRA)	7-175
ROR(16 位数据的循环右移)	7-30	设置电机控制(MC)	5-42
ROTC(循环表近路径循环移位控制)	6-122	设置状态闭锁(SLT)	7-173
RSET(切换文件寄存器号码)	7-290	十进制 ASCII 到 BCD 4 位数的转换(DABCD) .	7-193
RST-复位报警器(RST F)	5-32	十进制 ASCII 到 BCD 8 位数据的转换 (DDABCD)	7-193
RST-复位软元件(RST)	5-30		

十进制 ASCII 到 BIN 16 位数据的转换 (DABIN).....	7-188
十进制 ASCII 到 BIN 32 位数据的转换 (DDABIN).....	7-188
十六进制 ASCII 到 BIN 16 位数据的转换 (HABIN).....	7-191
十六进制 ASCII 到 BIN 32 位数据的转换 (DHABIN).....	7-191
十六进制 BIN 到 ASCII 的转换 (ASC)	7-225
时钟数据加法运算 (DATE+)	7-304
时钟数据减法运算 (DATE-)	7-306
时钟指令	2-42
实数数据	3-8
输出 (OUT)	5-18
输出 OFF 调用 (FCALL)	7-95
输出的脉冲转换 (DELTA)	5-38
输出取反 (FF)	5-36
输出指令 (OUT)	5-18
输出指令	2-6
数据处理指令	2-28
数据控制指令	2-40
数据以字节为单位分解 (WTOB)	7-71
数据以字节为单位链接 (BTOW)	7-72
数据转换指令	2-18
	6-57
数指令	5-18
刷新指令 (COM)	7-112
双字数据	3-5
顺控程序停止 (STOP)	5-50
顺序指令	2-4
算术运算指令	2-13
随机数据的分解 (NDIS)	7-66
随机数生成和序列更新 (RND, SRND)	7-264
索引寄存器的批量保存 (ZPUSH)	7-338
索引寄存器的批量恢复 (ZPOP)	7-338

[T]

TAN (T 浮点数数据的 TAN 运算)	7-246
TEST (位测试)	7-46
TO (特殊功能模块 1 字数据写运算)	7-143
TRACE (跟踪设置)	9-6
TRACER (跟踪复位)	9-6
TRUCK (数据跟踪)	10-6
TTMR (教学定时器)	6-117
特殊定时器 (STMR)	6-119
特殊格式故障检查 (CHKST, CHK)	7-161
特殊功能模块 1 字数据读运算 (FROM)	7-140
特殊功能模块 1 字数据写运算 (TO)	7-143
特殊功能模块 2 字数据读运算 (DFRO)	7-140

特殊功能模块 2 字数据写运算 (DTO)	7-143
特殊功能指令	2-38
梯形图块并行连接 (ORB)	5-7
梯形图块串行连接 (ANB)	5-7
调试和故障诊断指令	2-34
跳转到 END (GOEND)	6-99

[U]

UDCNT1 (计数器 1 相输入增大或减小)	6-110
UDCNT2 (计数器 2 相输入增大或减小)	6-115
UNI (16 位数据的 4 位 连接)	7-64
UNIRD (读模块信息)	9-2

[V]

VAL (字符串到 BIN 16 位的转换)	7-209
------------------------------	-------

[W]

WAND (16 位数据的逻辑乘积)	7-3
	7-5
WDT (复位看门狗定时器)	7-324
WOR (16 位数据的逻辑求和)	7-10
	7-12
WRITE (软元件数据写到站点)	8-27
WSUM (16 位数据的总数计算)	7-83
WTOB (以字节为单元的数据分解)	7-71
WTOW (以字节为单元的数据连接)	7-72
WXNR (16 位数据否定排它逻辑求和运算)	7-22
	7-26
WXOR (16 位数据排它或运算)	7-16
	7-18
外围软元件指令	2-43
网络刷新指令 (ZCOM)	8-6
位测试 (TEST/DTEST)	7-46
位处理指令	2-27
位软元件的批量复位 (BKRST)	7-48
位软元件的位数据指定	3-4
位软元件输出取反 (FF)	5-36
位软元件移位	5-40
位数据	3-2
位数据指定	3-4
文件寄存器 1 字节写 (ZRWRB)	7-331

[X]

XCH (32 位数据交换)	6-91
显示指令	2-33
相关编程手册	1-1
斜坡信号 (RAMP)	6-124
写路径参数 (RTWRITE)	8-103
写软元件数据到其它站点 (SWRITE)	8-34

写软元件数据到其它站点(ZNWR)	8-84
写时钟数据(DATAWR)	7-300
写数据到数据表(FIFW)	7-131
写数据到远程 I/O 站的 特殊功能模块(RTOP)	8-95
写数据到远程 I/O 站的 特殊功能模块(ZNTO)	8-72
循环表近路径循环移位控制(ROTC)	6-122
循环移位指令	2-26

[Y]

页改变(NOPLF, PAGE n)	5-52
移位指令	5-40
移位指令(顺序指令)	2-6
移位指令(应用指令)	2-27
应用指令	2-24
用于文件寄存器使用的设置文件(QDRSET)	7-294
用于注释的 LED 显示指令(LEDCL)	7-150
用于注释的文件设置(QCDSET)	7-294
与(AND)	5-2
与取反(ANI)	5-2

[Z]

ZCOM(网络刷新指令)	8-7
ZNFR(远程 I/O 站上的特殊功能 模块读数据)	8-67
ZNRD(其它站读软元件数据)	8-77
	8-81
ZNWR(写软元件数据到站)	8-84
	8-88
ZN 到(写数据到远程 I/O 站上的特殊功能模块)	8-72
ZONE(BIN 16 位数据的零区控制)	7-287
ZPOP(索引寄存器的批量恢复)	7-338
ZPUSH(索引寄存器的批量保存)	7-338
ZRRDB(文件寄存器中读 1 字节)	7-329
ZRWRB(文件寄存器 1 字节写)	7-331
载入	5-2
载入取反(LDI)	5-2
在程序文件之间的子程序调用(ECALL)	7-100
在程序文件之间输出 OFF 调用(EFCALL)	7-104
在数据表中插入数据(FINS)	7-137
整个梯形图的变址修改(IX, IXEND)	7-118
执行程序跟踪(PTRAEXE)	7-177
直接地址读(ADRSET)	7-333
直接输出的脉冲转换	5-38
指定数据	3-2
指令表	2-1

指令的执行条件	3-19
指令类型	2-1
指针分支(CJ)	6-96
中断程序掩码(IMASK)	6-100
中断禁止(DI)	6-100
中断允许(EI)	6-100
终止指令	2-7
转换	

BCD 4 位数据(BCD)	6-57
BCD 4 位数据到 BIN(BIN)	6-59
BCD 8 位数据(DBCD)	6-57
BCD 8 位数据到 BIN(DBIN)	6-59
BIN 16 位到 BIN 32 位(DBL)	6-65
BIN 16 位到格雷码(GRY)	6-67
BIN 32 位到 BIN 16 位(字)	6-66
BIN 32 位到格雷码(DGRY)	6-67
浮点数据到 BIN 16 位(INT)	6-63
浮点数据到 BIN 32 位(DINT)	6-63
子程序调用(CALL)	7-96
子集处理	3-15
字符串到 BIN 16 位数据的转换(VAL)	7-209
字符串到 BIN 32 位数据的转换(DVAL)	7-209
字符串到浮点数的转换(EVAL)	7-221
字符串中随机选择(MIDR)	7-232
字(BIN 16 位到 BIN 32 位数据的转换)	6-66
字符串查找(INSTR)	7-236
字符串长度检测(LEN)	7-201
字符串处理指令	2-35
字符串数据	3-9
字符串数据比较	6-8
字符串中随机选择替换(MIDW)	7-232
字符串转移(\$MOV)	6-82
字软元件的位复位(BRST)	7-44
字软元件的位置位(BSET)	7-44
字软元件位指定	3-3
字数据	3-3
最高和最低字节交换(SWAP)	6-95

[数字]

7 段解码(SEG)	7-60
8 位到 256 位解码(DECO)	7-56
16 位负转移(CML)	6-84
16 位数据的 4 位编组(DIS)	7-62
16 位数据的 4 位连接(UNI)	7-64
16 位数据的 n 位右移(SFR)	7-38
16 位数据的 n 位左移(SFL)	7-38
16 位数据的逻辑乘积(WAND)	7-3
16 位数据的逻辑求和(WOR)	7-10
	7-12

16 位数据的循环右移 (ROR, RCR)	7-30	32 位数据与或运算 (DXNR)	7-23
16 位数据的循环左移 (ROL, RCL)	7-32		7-27
16 位数据的总数计算 (WSUM)	7-83	32 位数据检查 (DSUM)	7-54
16 位数据的最大值搜索 (MAX)	7-75	32 位数据交换 (XCH)	6-91
16 位数据的最小值搜索 (MIN)	7-77	32 位死区控制 (DBAND)	7-284
16 位数据与或运算 (WXNR)	7-22	32 位排它或运算 (DXOR)	7-16
	7-26	32 位转移 (DMOV)	6-78
16 位数据检查 (SUM)	7-54	256 到 8 位的编码 (ENCO)	7-58
16 位数据交换 (XCH)	6-91		
16 位数据块转移 (FMOV)	6-89	[符号]	
16 位数据搜索 (SER)	7-50	\$+(连接字符串)	6-49
16 位死区控制 (BAND)	7-284		6-51
16 位异或运算 (WXOR)	7-16	\$MOV(字符串转移)	6-82
	7-18	*(BIN 16 位乘法和运算)	6-24
16 位转移 (MOV)	6-78	+(BIN 16 位加法运算)	6-16
32 位负转移 (DCML)	6-84		6-18
32 位数据的逻辑乘积 (DAND)	7-3	-(BIN 16 位减法运算)	6-16
	7-6		6-18
32 位数据的逻辑求和 (DOR)	7-10	/ (BIN 16 位除法运算)	6-26
	7-12	= (BIN 16 位数据比较)	6-2
32 位数据的循环右移 (DROR, DRCR)	7-34	>= (BIN 16 位数据比较)	6-2
32 位数据的循环左移 (DROL, DRCL)	7-36	> (BIN 16 位数据比较)	6-2
32 位数据的总数计算 (DWSUM)	7-85	< (BIN 16 位数据比较)	6-2
32 位数据的最大值搜索 (DMAX)	7-75	<= (BIN 16 位数据比较)	6-2
32 位数据的最小值搜索 (DMIN)	7-78	<> (BIN 16 位数据比较)	6-2

保修书

请在开始使用产品之前确认下列产品保修细节。

1. 免费保修期限和免费保修范围

在免费保修项目内，如果在产品的使用过程中，发现有任何异常或缺陷（下文中称为“故障”）是三菱电气公司的责任，此产品将无偿通过经销商或者三菱售后服务公司进行修理。

注意，如果产品修理是在海外的某个地点，一个孤岛上或者某个偏僻的地方，要收取派遣工程师所花的费用。

[免费保修期限]

产品的免费保修条款在订货日期或者交货到指定地点后一年内有效。

注意，在产品从三菱制造和出货后，产品的最大销售周期是六个(6)个月，并且最长免费保修期限是产品生产制造之后的十八(18)个月。维修部件的免费保修期限，在维修之前不能超过免费保修期限。

[免费保修范围]

- (1) 保修范围限制在规定的的使用状态、使用方法和使用环境等内的正常使用，这必须遵守在指令手册、用户手册和产品的警告标签上给出的关于使用条件和预防措施等的说明。
- (2) 即使在免费保修期限，对于下列情况，维修也是要收费的。
 1. 由于不正确的储藏或操纵，用户的粗心或疏忽而造成的故障。由用户的硬件或者软件设计而造成的故障。
 2. 由于用户对产品进行了未经三菱同意的修改等，而造成的故障。
 3. 当三菱产品被装配到用户的设备中时，按照法定的安全规范或工业标准来判断而应该对用户的设备进行功能或者结构测试而被用户拒绝，由此而造成的故障。
 4. 如果正确保养或替换在指令手册中指定的消耗性部件（电池、背光灯、保险丝等），就可以避免的故障。
 5. 由外部不可抗力，如火灾或不正常的电压等，引起的故障。由自然灾害，如地震、雷电、风灾和水灾引起的故障。
 6. 在从三菱的发货过程中，当前的科学技术标准无法预测的原因引起的故障。
 7. 任何其他既不是三菱公司也不是用户的责任而引起的故障。

2. 产品生产终止后的法律保修期限

- (1) 三菱公司在产品的生产被终止后的七(7)年内接收法律范围内的产品保修。

停止生产的产品将在三菱公司的技术公报中给出通知。

- (2) 生产停止后将不再进行产品供货(包括维修部件)。

3. 海外服务

海外维修将由三菱公司的当地海外 FA 中心接收。注意，每个 FA 中心的维修条件可能不同。

4. 保修义务中不包括的偶然损失和次要损失

不管免费保修的期限，三菱公司对下列情况负责赔偿：任何由于三菱公司责任的原因引起的损坏，偶然损失，由于三菱公司产品的失灵而给用户带来的利益损失，出乎三菱公司意愿的特殊原因引起的损坏和次要损坏，以及对事故的赔偿；而对于非三菱公司的产品的损坏和其他责任引起的赔偿，三菱公司不负责赔偿。

5. 产品说明书中的更改

对于目录，手册或技术文档中说明书的更改，恕不提前通知。

6. 产品应用范围

- (1) 在使用三菱公司 MELSEC 可编程控制器的过程中，使用条件应该满足以下要求：即使在可编程逻辑控制器设备中发生任何问题或故障，此应用不会引起主要事故；对于任何问题或故障，必须在设备外系统地提供备份和故障安全功能。
- (2) 三菱公司的通用可编程逻辑控制器是按照一般工业应用进行设计和制造的。因此，对于那些公众可能受到影响的应用领域，例如由个别电力公司建设的核电站和其他电站，以及需要特殊品质保证系统的应用领域，例如日本铁路公司或安全防卫厅等，应该不包括在可编程逻辑控制器的应用范围之内。

注意，即使是在这些应用领域，如果用户保证此应用场合不受限制，并且不需要特殊品质，则这些应用可以使用三菱公司的可编程逻辑控制器。

当考虑将三菱公司的可编程逻辑控制器使用在航空、医学应用领域、铁路、焚化和燃料设备、人造运输设备、用于娱乐和消遣的装置以及安全设备时，由于在这些应用场合中人的生命或财产会受到极大影响，并且对安全和控制系统有非常高的稳定性要求，因此请联系三菱公司并就特殊的要求进行讨论。

QCPU (Q模式) / QnACPU

编程手册 (公共指令)



菱电自动化(上海)有限公司
RYODEN AUTOMATION (SHANGHAI) LTD.
菱电集团及三菱电机附属机构

地址: 上海漕宝路103号自动化仪表城5号楼1~3层
电话: 021-64753228 传真: 021-64846996
邮编: 200233
网址: www.ryoden-automation.com.cn

书号	SH (NA) -080450 CHN
印号	RAS-QPLC-CI-C

内容如有更改, 恕不另行通知